# COMPUTERIZED AERODYNAMIC OPTIMIZATION
# OF
# AIRCRAFT PROPELLERS

Robert Linford Shaw

United States
Naval Postgraduate School

# THESIS

## COMPUTERIZED AERODYNAMIC OPTIMIZATION OF AIRCRAFT PROPELLERS

by

Robert Linford Shaw

June 1970

*This document has been approved for public release and sale; its distribution is unlimited.*

1134516

# Computerized Aerodynamic Optimization
# of
# Aircraft Propellers

by

Robert Linford Shaw
Lieutenant (junior grade) United States Navy
B. S., Purdue University, 1969

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June, 1970

STGRADUATE S
, CALIF. 93940

## ABSTRACT

The objective of this thesis was to develop a practical computer system for use of empirical data in the aerodynamic optimization of aircraft propellers. The system was designed for use with the IBM 2741 on-line computer terminal. This program provides instructions to the operator during execution, and allows interaction by the operator for input and alteration of data, and for program instructions.

The Lockheed P-3C aircraft was chosen as the subject for test and evaluation of the program. The currently operational propeller of this aircraft was tested to compare the program's prediction of aircraft performance against flight test information. An attempt was then made to select a propeller which would provide better performance under the same constraints as those imposed in design of the operational propeller.

# TABLE OF CONTENTS

I. INTRODUCTION . . . . . . . . . . . . . . . . . . . . . . . 9

II. PROGRAM ORGANIZATION AND OPERATION . . . . . . . . . . . 12

- A. GENERAL ORGANIZATION . . . . . . . . . . . . . . . 12
- B. INPUT ROUTINE . . . . . . . . . . . . . . . . . . . 14
- C. OPTIMIZATION ROUTINE . . . . . . . . . . . . . . . 23
- D. PERFORMANCE EVALUATION ROUTINE . . . . . . . . . . 31

III. PROGRAM USAGE . . . . . . . . . . . . . . . . . . . . . 43

- A. OPTIMIZATION OF PROPELLERS FOR GIVEN AIRCRAFT . . 43
- B. EVALUATION OF A GIVEN PROPELLER CONFIGURATION . . 50

IV. EXAMPLE PROBLEM: LOCKHEED P-3C . . . . . . . . . . . . 51

- A. EVALUATION . . . . . . . . . . . . . . . . . . . . . 51
- B. OPTIMIZATION OF P-3 PROPELLER . . . . . . . . . . . 53

V. EVALUATION OF PROJECT SUCCESS . . . . . . . . . . . . . 55

VI. PROBLEMS ENCOUNTERED AND AREAS FOR FUTURE RESEARCH . . 56

APPENDIX A: COMPUTER PROGRAM . . . . . . . . . . . . . . . 58

APPENDIX B: SAMPLE COMPUTER OUTPUT . . . . . . . . . . . . 90

LIST OF REFERENCES . . . . . . . . . . . . . . . . . . . . 102

INITIAL DISTRIBUTION LIST . . . . . . . . . . . . . . . . 103

FORM DD 1473 . . . . . . . . . . . . . . . . . . . . . . . 105

# LIST OF TABLES

1. P-3C DATA SHEET . . . . . . . . . . . . . . . . . . . . . . 52

2. COMPARISON OF FLIGHT TEST DATA AND PREDICTED PERFORMANCE OF P-3C AIRCRAFT . . . . . . . . . . . . . . . 53

3. COMPARISON ON OPERATIONAL P-3C PROPELLER AND SAMPLE OPTIMUM PROPELLER . . . . . . . . . . . . . . . . . 54

4. PREDICTED PERFORMANCE COMPARISON BETWEEN OPERATIONAL P-3 PROPELLER AND SAMPLE OPTIMUM . . . . . . . 54

## LIST OF FIGURES

1. GENERAL PROGRAM ORGANIZATION . . . . . . . . . . . . . . . . . . 13

2. INPUT ROUTINE FLOW DIAGRAM . . . . . . . . . . . . . . . . . . . 15

3. STARTING POINT ROUTINE . . . . . . . . . . . . . . . . . . . . . 21

4. AEROPRODUCTS PRELIMINARY PROPELLER SELECTION CHART . . . . . 22

5. OPTIMIZATION ROUTINE . . . . . . . . . . . . . . . . . . . . . . 24

6. EXPLORATORY STEP OF OPTIMIZATION ROUTINE . . . . . . . . . . . 25

7. SEARCH PROCEDURE FOR NUMBER OF BLADES . . . . . . . . . . . . 30

8. AIRCRAFT PERFORMANCE EVALUATION ROUTINE . . . . . . . . . . . 33

9. CALCULATION OF MAXIMUM VELOCITY AND CRUISE VELOCITY . . . . 39

10. CALCULATION OF CRUISE RATE . . . . . . . . . . . . . . . . . 42

# I. INTRODUCTION

In 1902, the Wright brothers, planning to adapt a theory used in the design of marine propellers, conducted a literature search in the Dayton, Ohio, public library. They found no theory, but only empirical data. A general theory of aircraft propeller performance was not forthcoming until 1929 in a paper presented to the Royal Society of London by Goldstein, [Ref. 1]. This "vortex theory" developed by Goldstein has since been revised and amplified [Ref. 2], but is still cumbersome and somewhat inaccurate for use in real propeller design.

Many wind tunnel tests were conducted on aircraft propellers before, during, and immediately following World War II, notably by Hartman and Biermann for the N.A.C.A. [Ref. 3]. Since that time the advent of the jet engine has caused a loss of interest in propeller research by all except a few propeller manufacturers.

Hence, even in 1970, aircraft propellers are still being designed by methods not much more sophisticated than those employed by the Wright brothers in 1902. A general theory exists which is limited in its applicability. Great volumes of empirical data, much of it useless in practice, has been generated. Still, however, the propeller designer basically searches through empirical charts for an aerodynamically optimum propeller configuration.

Hamilton Standard Division of United Aircraft, probably the leader in the field of propeller design, has estimated sixteen man-hours of this type of work for each propeller design. Some use was being made of modern digital computers, however, to choose the best design after the range of possibilities had been narrowed by the manual process. The programs used were reportedly based upon Goldstein's vortex theory of 1929.

The results of the extensive research and the general theories, in spite of the lack of sophistication, have been remarkable. The Wright brother's propeller, designed by a theory of their own, had an efficiency of about 66%. Modern propellers can be expected to exhibit efficiencies above 90%, in some cases.

Judging from the history of the past seventy years, it seems reasonable to assume that empiricism will continue to dominate the propeller field in the foreseeable future. Therefore, it is the purpose of this thesis to make the procedure of designing an aerodynamically optimum propeller a little easier, and less time consuming, by employing the on-line digital computer systems now available.

The result of this thesis was a computer program, PROPOP, designed for use with the IBM 2741 on-line computer console. PROPOP asks questions concerning the aircraft, engines, environment, constraints and optimization criteria. The required information is supplied by the operator through the communications console.

PROPOP was designed with the intent that it contain all the instructions needed by the operator to manipulate the program successfully. Parts II and III of this thesis serve as a user's manual, in case more detailed instructions on program use are required, or in the event the user wishes to modify the program in some way. Theoretically, no knowledge of FORTRAN or internal operation of the program is required for use of PROPOP, beyond the ability to use the IBM 2741 console (or similar system). Once execution of PROPOP begins, all instructions for its operation are provided by the program itself.

The advantages of PROPOP lie in the integration of propeller efficency determination and aircraft performance evaluation. Present programs

apparently begin with assumed operating conditions for horsepower, altitude and velocity, then optimize propeller efficiency for these conditions. However, after this propeller is determined, it may be found that the aircraft does not actually operate at the assumed conditions. Consequently, the propeller is no longer optimum. The designer is then forced either to ignore this fact or to choose new assumed operating conditions and repeat the optimization process. In effect, PROPOP automatically chooses the latter of these two alternatives. Much flexibility is also incorporated into the system by allowing the operator to select the relative importance of the various aircraft performance characteristics, and by allowing him to impose constraints on aircraft performance.

# II. PROGRAM ORGANIZATION AND OPERATION

## A. GENERAL ORGANIZATION

As shown in Fig. 1, the complete program consists of three primary routines:

1. Input Subroutine
2. Optimization Subroutine
3. Performance Evaluation Subprogram

Immediately upon execution of the program, propeller performance data are read through the off-line card reader. These data are internal to the program. A message is typed on the operator's console when all data have been read. A brief explanation of the program is then printed, followed by a list of steps the operator must perform.

The Input Routine then begins. This routine asks questions of the operator about the aircraft, engines, etc., for which the propeller is to be optimized. The operator responds by typing in the required information and the program proceeds to the next question.

PROPOP operates in two modes. If the operator completely specifies the propeller during operation of the Input Routine, the program operates in the evaluation mode. In this mode the program proceeds from the Input Routine to the Performance Evaluation Routine (dashed line, Fig. 1) and evaluates the aircraft performance with the given specified propeller. However, should the operator leave any or all the propeller parameters unspecified, the program operates in the optimization mode. In this mode, after completion of the Input Routine, the program proceeds to the Optimization Routine and optimizes for the unspecified parameters (solid line, Fig. 1). The Optimization Routine repeatedly calls upon the Performance

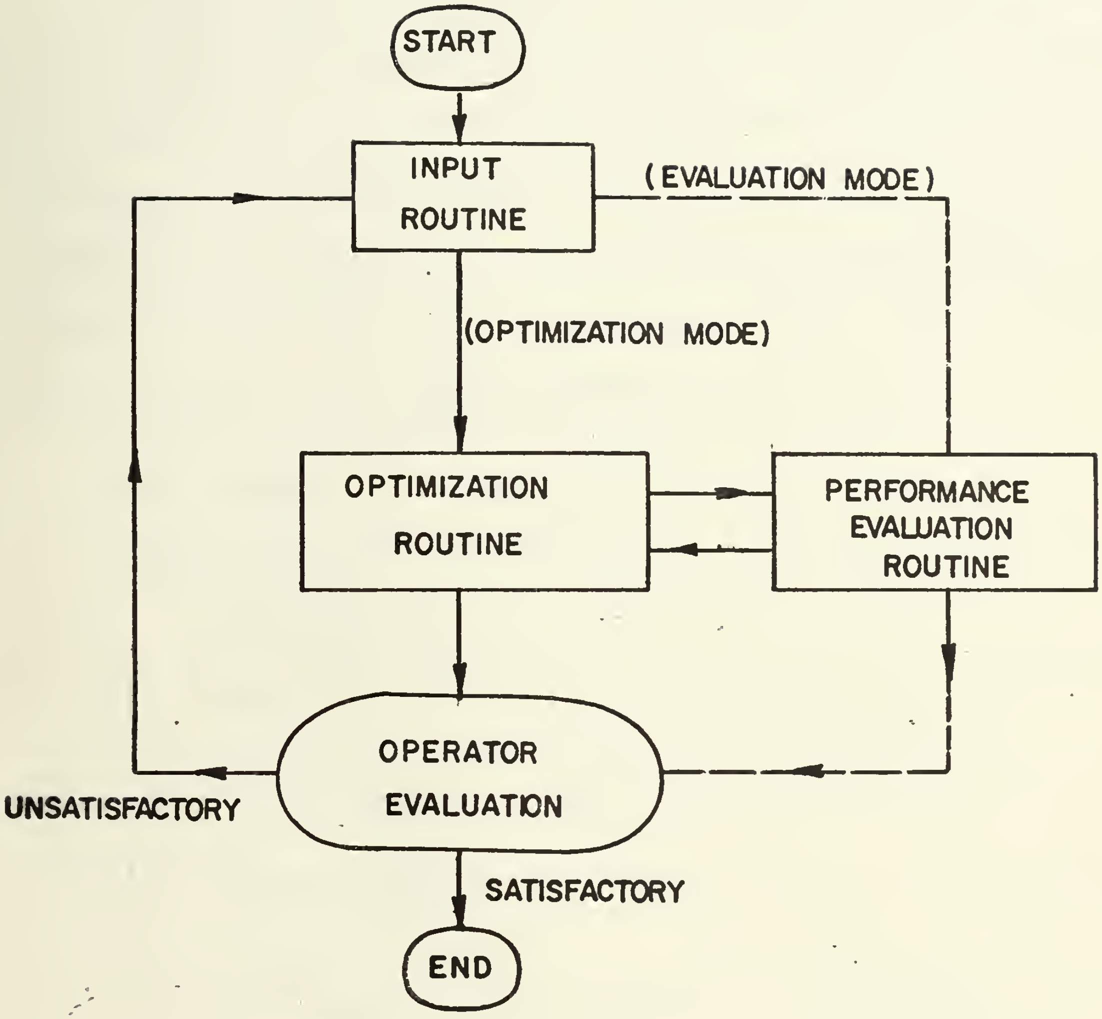

FIGURE I

GENERAL PROGRAM FLOW DIAGRAM.

Evaluation Routine for information needed in the optimization procedure. At its completion, the Optimization Routine prints the optimum propeller parameters as well as the aircraft performance with this propeller.

The next step in the procedure, regardless of the mode of operation, is evaluation of the output by the operator. At this point the operator may either terminate the program or he may return to the Input Routine and change any of the information that he has previously supplied. The usual action would probably be to change the constraints imposed on the aircraft or the propeller, or to change the priority given to any particular portion of the aircraft's mission.

## B. INPUT ROUTINE

The Input Routine, Fig. 2, consists of seven input steps:

1. Aircraft data
2. Engine data
3. Aircraft environment and anticipated performance
4. Aircraft performance constraints
5. Propeller parameter constraints
6. Optimization criteria
7. End of routine (begin evaluation or optimization)

Each of the first six steps requires the operator to furnish data to the program by typing it into his terminal. Step 7 is only a reference point within the routine so that the operator can refer to the end of the routine by calling for Step 7. The broken lines without arrows in Fig. 2 indicate that the operator may skip from one step to another, either forward or backward, within the Input Routine. For instance, Step 4 and Step 6 are not required if the program is to operate in the evaluation mode, so these steps may be omitted altogether.

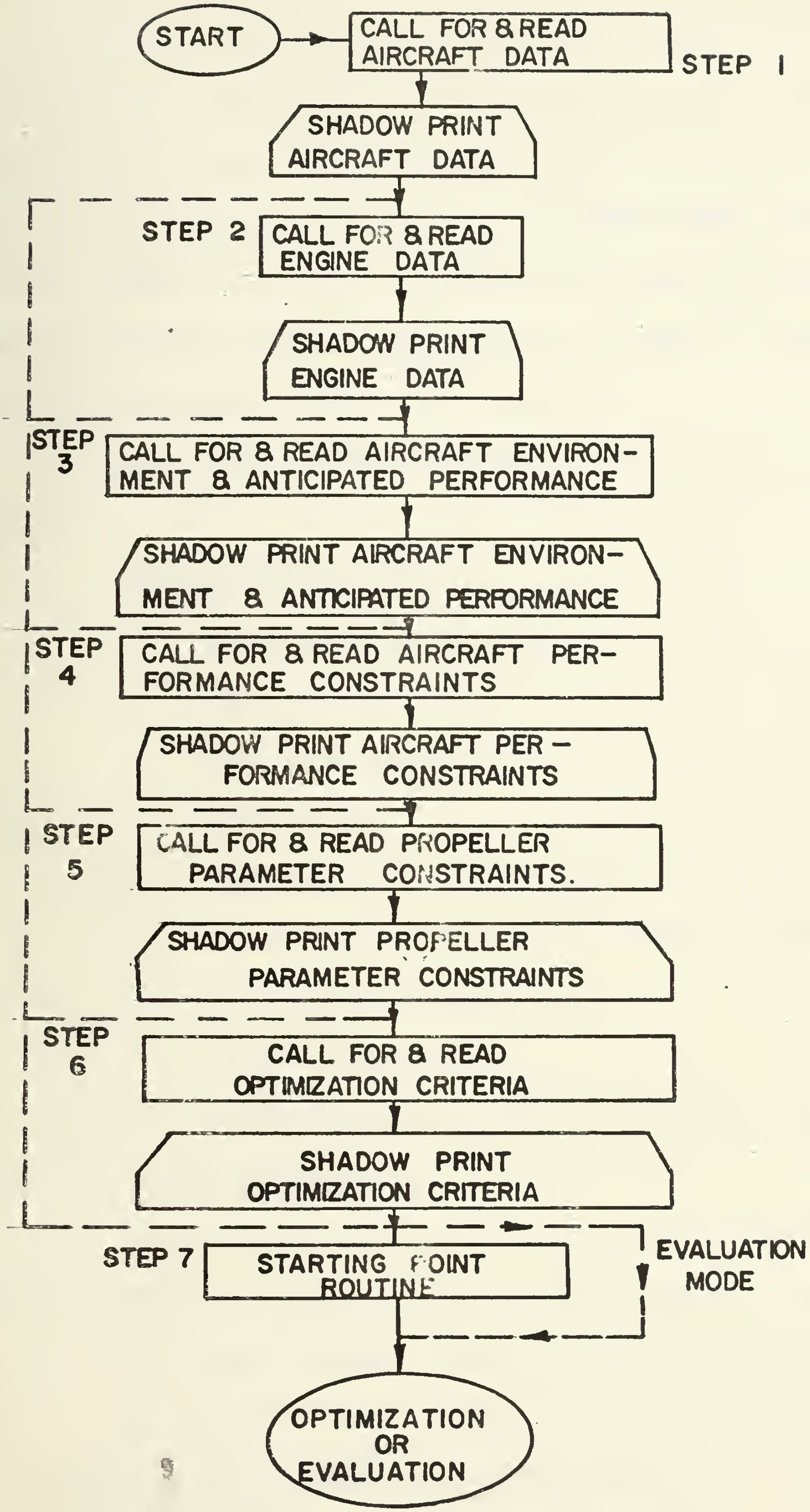
START
CALL FOR & READ AIRCRAFT DATA
STEP 1
SHADOW PRINT AIRCRAFT DATA
STEP 2
CALL FOR & READ ENGINE DATA
SHADOW PRINT ENGINE DATA
STEP 3
CALL FOR & READ AIRCRAFT ENVIRONMENT & ANTICIPATED PERFORMANCE
SHADOW PRINT AIRCRAFT ENVIRONMENT & ANTICIPATED PERFORMANCE
STEP 4
CALL FOR & READ AIRCRAFT PERFORMANCE CONSTRAINTS
SHADOW PRINT AIRCRAFT PERFORMANCE CONSTRAINTS
STEP 5
CALL FOR & READ PROPELLER PARAMETER CONSTRAINTS.
SHADOW PRINT PROPELLER PARAMETER CONSTRAINTS
STEP 6
CALL FOR & READ OPTIMIZATION CRITERIA
SHADOW PRINT OPTIMIZATION CRITERIA
STEP 7
STARTING POINT ROUTINE
EVALUATION MODE
OPTIMIZATION OR EVALUATION

FIGURE 2

INPUT ROUTINE FLOW DIAGRAM

In each of the first six steps, PROPOP asks the operator for information required by the program. The operator then types these data into his console. At the end of each step, all the data supplied by the operator are printed on the terminal so that the operator may check it for accuracy. Should any data be incorrect, the operator simply returns to the beginning of that step and begins again to answer the questions asked by the program, typing in the correct data.

The information called for by each step in the Input Routine is as follows:

1. Step 1 (Aircraft Data)
    a. Design gross weight (lbs.)
    b. Wing area (sq. ft.)
    c. Number of engines
    d. Aspect ratio
    e. Parasitic drag coefficient (takeoff configuration)
    f. Parasitic drag coefficient (cruise configuration)
    g. Wing efficiency factor (e)
    h. Fuel capacity (lbs.)
    i. Takeoff speed (kts. TAS)
2. Step 2 (Engine Data)

    Engine data required by this step are determined by whether the aircraft uses reciprocating or turbine engines.

    a. Type of Engine (Reciprocating or Turbine)
    b. Reciprocating Engine
        (1) Takeoff brake horsepower
        (2) Engine RPM during takeoff
        (3) Brake horsepower during climb

(4) Engine RPM during climb

(5) Brake horsepower during cruise

(6) Engine RPM during cruise

(7) Brake horsepower at maximum aircraft speed

(8) Engine RPM at maximum aircraft speed

(9) Fuel consumption (per engine) during cruise (lbs/hr)

c. Turbine Engine

(1) Engine operating RPM

(2) Shaft horsepower (per engine) at cruise altitude and cruise power, for anticipated cruise speed and for 50 kts. above this speed

(3) Shaft horsepower (per engine) at maximum power setting (power for takeoff and maximum speed) and at 10,000 ft. MSL

(4) Sea level shaft horsepower (per engine) for maximum power setting at zero, 100, and 400 kts. TAS

(5) Sea level jet thrust (per engine) for maximum power setting at zero, and 100 kts. TAS

(6) Sea level jet thrust for cruise power setting at zero and 100 kts. TAS

(7) Fuel consumption (lbs/hr) per engine for cruise power and altitude, at anticipated cruise speed and 50 kts. above this speed

(8) Limiting shaft horsepower for continuous operation

All data requested for reciprocating engines should correspond to the referenced aircraft operating conditions. For example, "cruise horsepower" should be given for cruise RPM, cruise altitude and anticipated cruise speed. All conditions are specified above for turbine engines.

All engine data are for the installed engine. Shaft horsepower (Item 4) for the turbine engine should be entered without regard to maximum allowable SHP due to gear box restrictions, etc. In other words, SHP's in Item (4) above may be greater than maximum allowable SHP in practice. This is necessary because of the interpolation process employed by the program. However, SHP's greater than the maximum allowable SHP are not actually used in calculating aircraft performance.

3. Step 3 (Aircraft Environment and Anticipated Performance

a. Takeoff altitude

b. Cruise altitude

c. Altitude for maximum speed calculations

d. Anticipated speed for best climb rate (TAS)

e. Anticipated cruise speed (kts. TAS)

f. Anticipated maximum speed (kts. TAS)

The altitudes specified above are those at which aircraft performance is calculated and are reference altitudes for much of the engine data requested in Step 2. The anticipated speeds are again reference points for engine data, but are also starting points for the iterative process employed by the computer to calculate aircraft performance.

4. Step 4 (Aircraft Performance Constraints)

This step is required only for optimization of a propeller, not for simple evaluation of a given propeller.

a. Maximum allowable takeoff distance (ft)

b. Lowest allowable maximum climb rate (ft/min)

c. Minimum allowable range at cruise conditions (n. mi.)

d. Minimum allowable endurance at cruise conditions (hrs)

e. Minimum allowable cruise speed (kts. TAS)

f. Lowest allowable maximum speed (kts. TAS)

5. Step 5 (Propeller Constraints)

   a. Maximum blade diameter for structural or ground clearance (ft)

   b. Number of blades (2, 3, or 4)

   c. Gear ratio (engine RPM/prop RPM)

   d. Blade diameter (ft)

   e. Activity factor (80-220 except for 2 blades, then 80-120)

   f. Integrated lift coefficient (.15-.7 or .3-.7 for 2 blades)

Although this step is required for both the evaluation and the optimization mode of PROPOP, only the first item (maximum blade diameter) must be specified. Any or all of the other items may be left open to optimization by typing zero (0.) after the request for that particular item. If any of these items is left unspecified in this manner, the program automatically operates in the optimization mode. Should all the items be specified, the program operates in the evaluation mode. The limits of the data currently available to the program are indicated above.

6. Step 6 (Optimization Criteria)

   This step is required only for the optimization mode.

   a. Value of importance of short takeoff distance

   b. Value of importance of climb rate

   c. Value of importance of range

   d. Value of importance of endurance

   e. Value of importance of cruise speed

   f. Value of importance of maximum speed

"Values of Importance" are numbers from zero to ten indicating the

relative value of a particular aircraft performance characteristic to the overall aircraft mission. The absolute value of these numbers is not critical, only the relative values. Where more than one performance characteristic is given a non-zero value, the Optimization Routine stresses the characteristics in the order of their values, the highest value receiving highest priority.

The "Starting Point Routine" (Fig. 3) is designed for use in the optimization mode of PROPOP. Its purpose is to make an intelligent guess at the optimum propeller configuration before the actual optimization procedure begins. This is necessary for three reasons. First, the Optimization Routine must begin somewhere and the closer it begins to the final optimum the better. Secondly, the procedure must start within the range of propeller data available. The optimization procedure itself forces the propeller to lie within the range of data after the procedure begins. The third, and probably most important reason, is that the first propeller configuration chosen should be one with which the aircraft can fly. For example, it is difficult for the program to determine which propeller parameter to change for an improvement in cruise speed if the aircraft cannot even takeoff with the starting propeller. In such a case PROPOP attempts to optimize climb rate in order to find a propeller with which the aircraft will fly. This procedure is, however, subject to possible failure.

The basis of the Starting Point Routine is the Aeroproducts Preliminary Propeller Selection Chart, a generalized copy of which is shown in Fig. 4. The data in this chart are included in PROPOP and have the effect of introducing an "experience factor" into the Starting Point Routine.

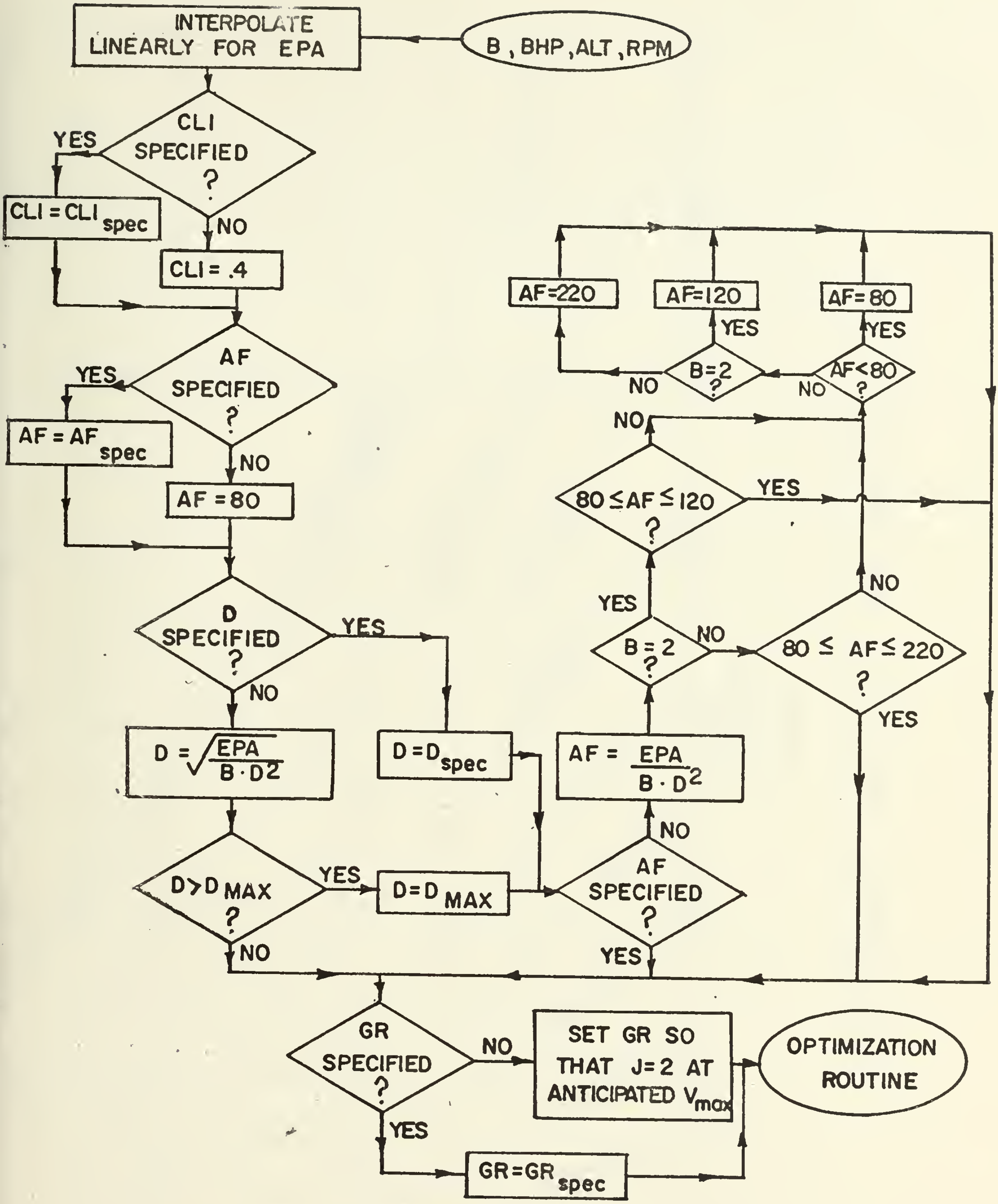

FIGURE 3

STARTING POINT ROUTINE

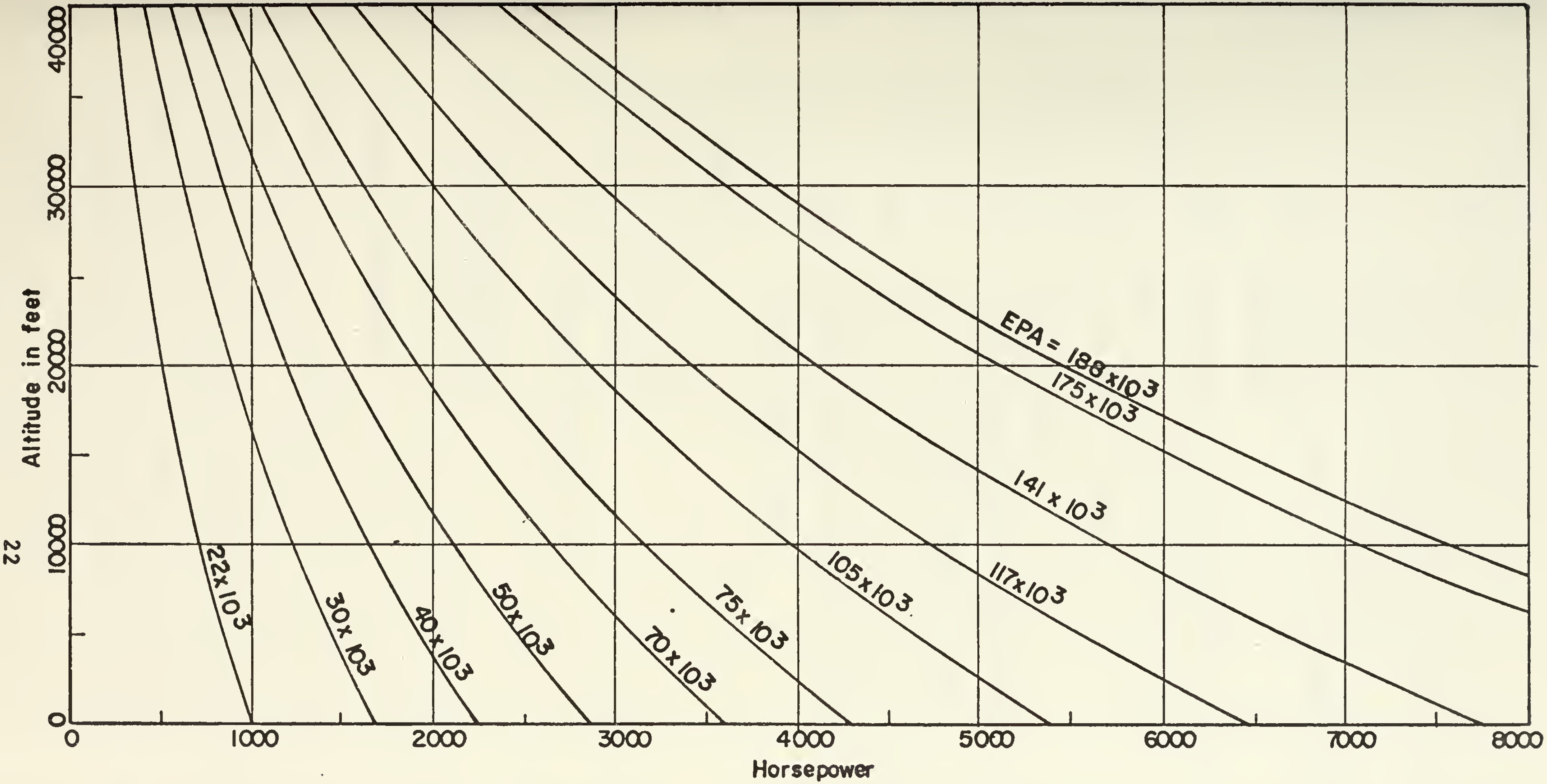

FIGURE 4

AEROPRODUCTS PROPELLERS — PRELIMINARY SELECTION

Explanation of terms in Fig. 3:

BHP - brake/shaft horsepower; horsepower specified for maximum velocity for reciprocating engine, maximum continuous horsepower for turbine engine

ALT - altitude at which maximum aircraft speed is to be evaluated

RPM - engine RPM at maximum aircraft velocity

B - number of propeller blades per engine

AF - blade activity factor

D - blade diameter

CLI - blade integrated lift coefficient

GR - gear ratio (engine RPM/prop RPM)

EPA - effective propeller area (B x AF x $D^2$) from Fig. 4.

DMAX- maximum allowable propeller diameter

J - advance ratio (v/nd)

where:

v = TAS, ft/sec

n = propeller RPS

D = Propeller diameter, ft.

## C. OPTIMIZATION ROUTINE

The optimization procedure is based on a pattern search technique developed by Hooke and Jeeves [Ref. 4, 5]. The subroutine DIRECT, incorporated into the IBM System/360 Subroutine Library in 1966 was used as the basic optimizer. It is this procedure which is outlined in Figs. 5 and 6. This subroutine is, however, an unconstrained minimization program. For this reason the routine had to be heavily modified so that it could handle the highly constrained propeller optimization problem.

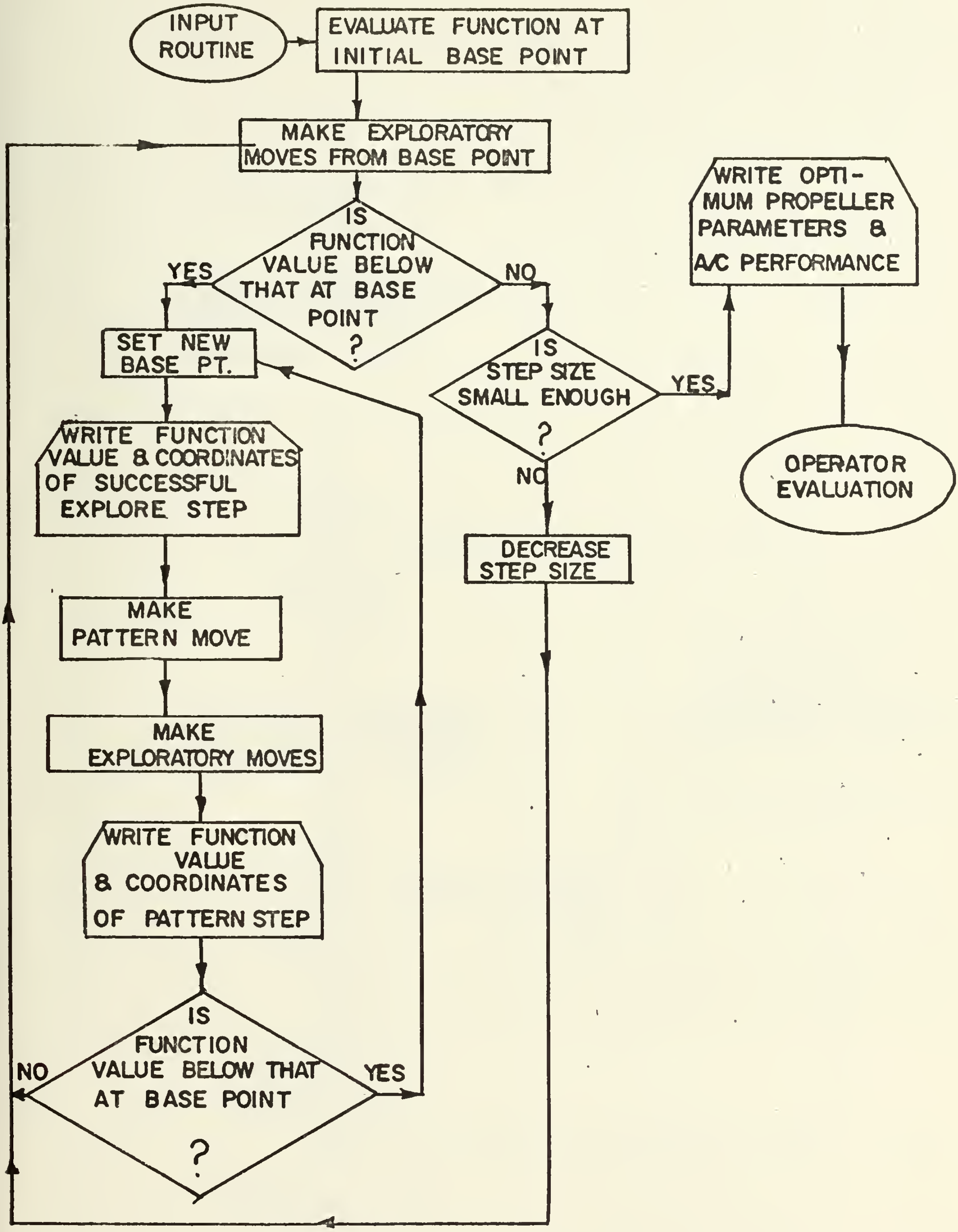
INPUT ROUTINE
EVALUATE FUNCTION AT INITIAL BASE POINT
MAKE EXPLORATORY MOVES FROM BASE POINT
IS FUNCTION VALUE BELOW THAT AT BASE POINT ?
YES
NO
SET NEW BASE PT.
IS STEP SIZE SMALL ENOUGH ?
YES
NO
WRITE OPTIMUM PROPELLER PARAMETERS & A/C PERFORMANCE
OPERATOR EVALUATION
WRITE FUNCTION VALUE & COORDINATES OF SUCCESSFUL EXPLORE STEP
DECREASE STEP SIZE
MAKE PATTERN MOVE
MAKE EXPLORATORY MOVES
WRITE FUNCTION VALUE & COORDINATES OF PATTERN STEP
IS FUNCTION VALUE BELOW THAT AT BASE POINT ?
NO
YES

FIGURE 5

OPTIMIZATION ROUTINE

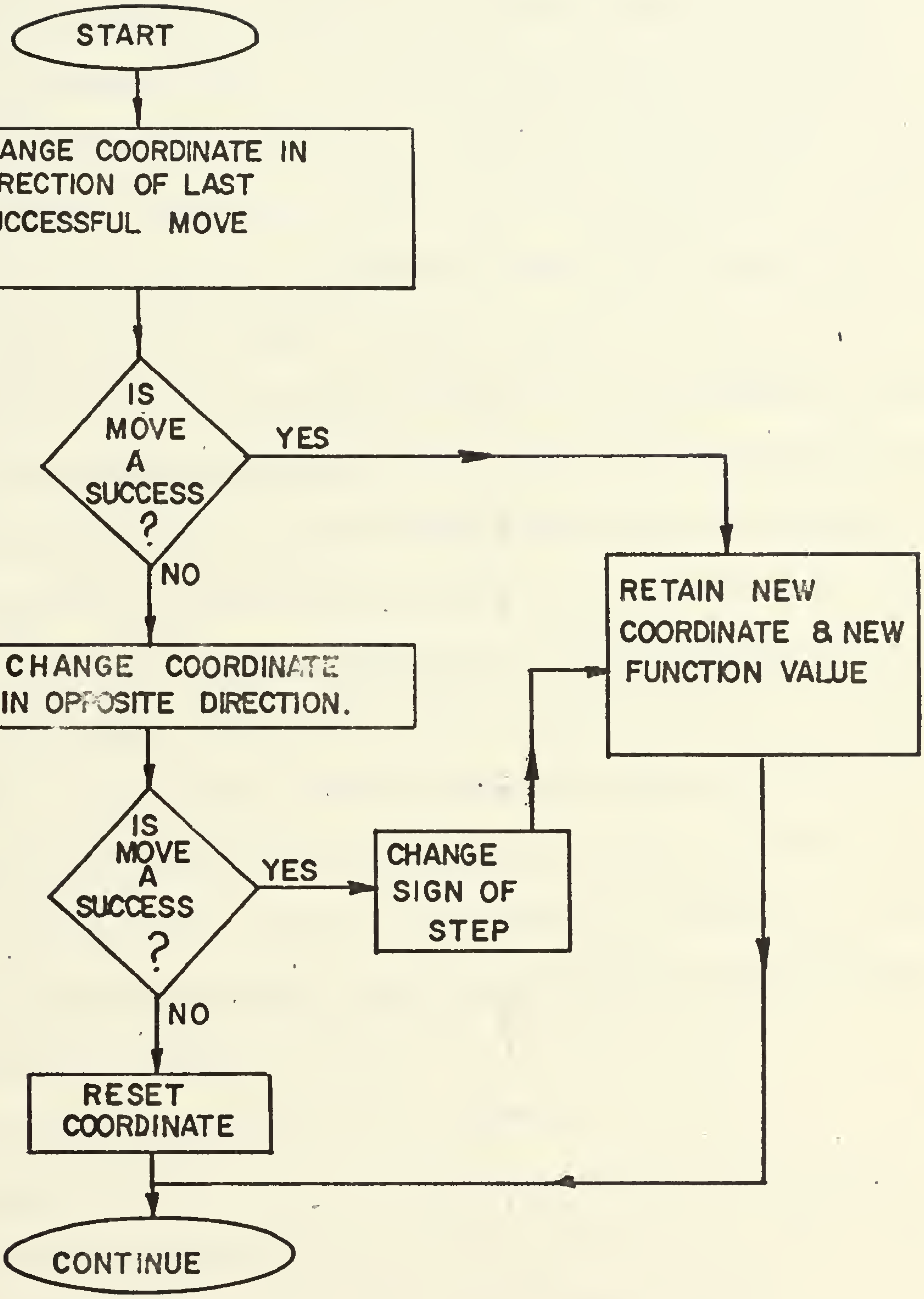
START
CHANGE COORDINATE IN DIRECTION OF LAST SUCCESSFUL MOVE
IS MOVE A SUCCESS ?
YES
NO
RETAIN NEW COORDINATE & NEW FUNCTION VALUE
CHANGE COORDINATE IN OPPOSITE DIRECTION.
IS MOVE A SUCCESS ?
YES
CHANGE SIGN OF STEP
NO
RESET COORDINATE
CONTINUE

FIGURE 6

EXPLORATORY STEP OF OPTIMIZATION ROUTINE.

Basically, pattern search is a numerical trial-and-error procedure closely akin to the better known "gradient search" methods [Ref. 6]. A fundamental difference, however, is that pattern search, unlike gradient methods, does not attempt to follow the path of steepest descent to the minimum of the function value. It is this very property which often foils gradient methods when they encounter a "ridge" or "steep valley" in the objective function. The pattern search begins by exploring the region of the starting (base) point by incrementing each variable separately in search of a function value which is better (lower in this case) than the base point (Fig. 6). After having explored in all the separate variable directions, the program will have normally found a better function value than the base point value. It then assumes that further decreases in the function value may be made in this same direction from the base point (Fig. 5). The program has now established a "pattern" and will continue to make pattern moves in this direction until a move fails to produce an improvement. At this point another exploratory sequence is performed and the procedure begins again. If all exploratory steps also fail to produce an improvement, the step size is decreased to increase the definition of the functional surface. The program terminates when the step size falls below a prescribed value.

Modification of this subroutine included built-in "stops" to insure that the search procedure does not leave the bounds of the available data. Other modifications allowed the operator to specify any of the variables so that the search procedure does not change these variables. The specified variables are known as equality constraints because they are always equal to a set value. Constraints which specify a range or a limit of

acceptable values of the variables, such as the data limits, are known as inequality constraints.

Another set of important inequality constraints are the aircraft performance constraints. These constraints were accomplished by modifications of the Performance Evaluation Routine rather than the optimization procedure itself. If at any time the propeller variables being tested by the Optimization Routine result in aircraft performance which is unacceptable by the operator's prescribed standards, a false function value is assigned for that trial. This false function value is designed so that it decreases very rapidly as the "boundary of acceptance" of the aircraft performance is approached.

In order to understand the operation of this technique more clearly, it is necessary to have an explanation of the way in which a functional value of aircraft performance is derived. The technique employed by PROPOP is as follows. For a given propeller configuration the performance of the aircraft is evaluated by the Evaluation Routine. A functional value (which has been derived by the author) is then assigned. This value has the form:

$$F = \sum_i [(VOI)_i \cdot P_i + N_i \cdot P_i' ]$$

where

$F$ is the functional value

$i$ is an index denoting one of the performance criteria (takeoff distance, maximum speed, etc.)

$(VOI)_i$ is the value of importance assigned by the operator to the $i^{th}$ performance criterion

$P_i$ is a normalized measure of the $i^{th}$ performance criterion which always decreases as the $i^{th}$ performance criterion improves

($P_i$ is of order 1)

$N_i$ is simply an indicator which equals zero if the $i^{th}$ criterion is acceptable and equals one if the $i^{th}$ criterion is unacceptable

$P_i'$ is another measure of the $i^{th}$ performance criterion which decreases as the $i^{th}$ criterion improves, but $P_i'$ is of order 100 or greater.

The Optimization Routine seeks to reduce the value of F, so as long as each performance criterion meets the constraints imposed by the operator, the routine attempts to decrease the value of $P_i$ and consequently improve the $i^{th}$ performance criterion. Because all $P_i$ are of order one, a high value of $(VOI)_i$ gives more weight to the $i^{th}$ criterion in the overall function. However, if one or more of the constraints is not satisfied, then $N_i$ equals one, which allows $P_i'$ to dominate the other terms of the function. The optimizer immediately goes to work on improving the $i^{th}$ criterion until the boundary of acceptance is crossed, at which time the routine continues to optimize the $P_i$ values giving preference to higher values of importance.

Five variables are recognized by the Optimization Routine. These are:

1. Number of blades
2. Gear ratio (engine RPM/prop RPM)
3. Propeller diameter
4. Blade activity factor
5. Blade integrated lift coefficient

Of these five, the first is a discrete variable, while the other four are

continuous variables. In other words, there can be only an integer number of blades, but the other variables may take on fractional values.

Discrete variables present a problem for numerical optimization routines. There are two widely used methods of treating this problem. Probably the most obvious method is to search for a local optimum with each possible value of the discrete variable, separately, finally choosing that value for which the best local optimum is found. The second is to treat the discrete variable as a continuous variable, allowing it to have fractional values, and then rounding off to the nearest integer value.

Both methods have advantages and disadvantages. The first approach would obviously be very time consuming if the discrete variable had many possible values. However, it would eventually find the correct optimum. The second method would be faster, but the rounding off procedure could easily be fooled. In practice, with a large range of possible values for the discrete variable, a combination of the two methods may be used. The continuous variable approach could be employed to narrow the choice to a few values of the variable, each of which could then be searched separately.

In the propeller optimization problem, data were available only for 2, 3, and 4-blade propellers, so there are only three possible values of the discrete variable. The individual search approach was therefore employed in the Optimization Routine. Figure 7 outlines this procedure. Unless the number of blades is specified by the operator, the routine first conducts a search with a 4-blade propeller, followed by a 3-blade search. If the 3-blade optimum is worse than the 4-blade optimum, the

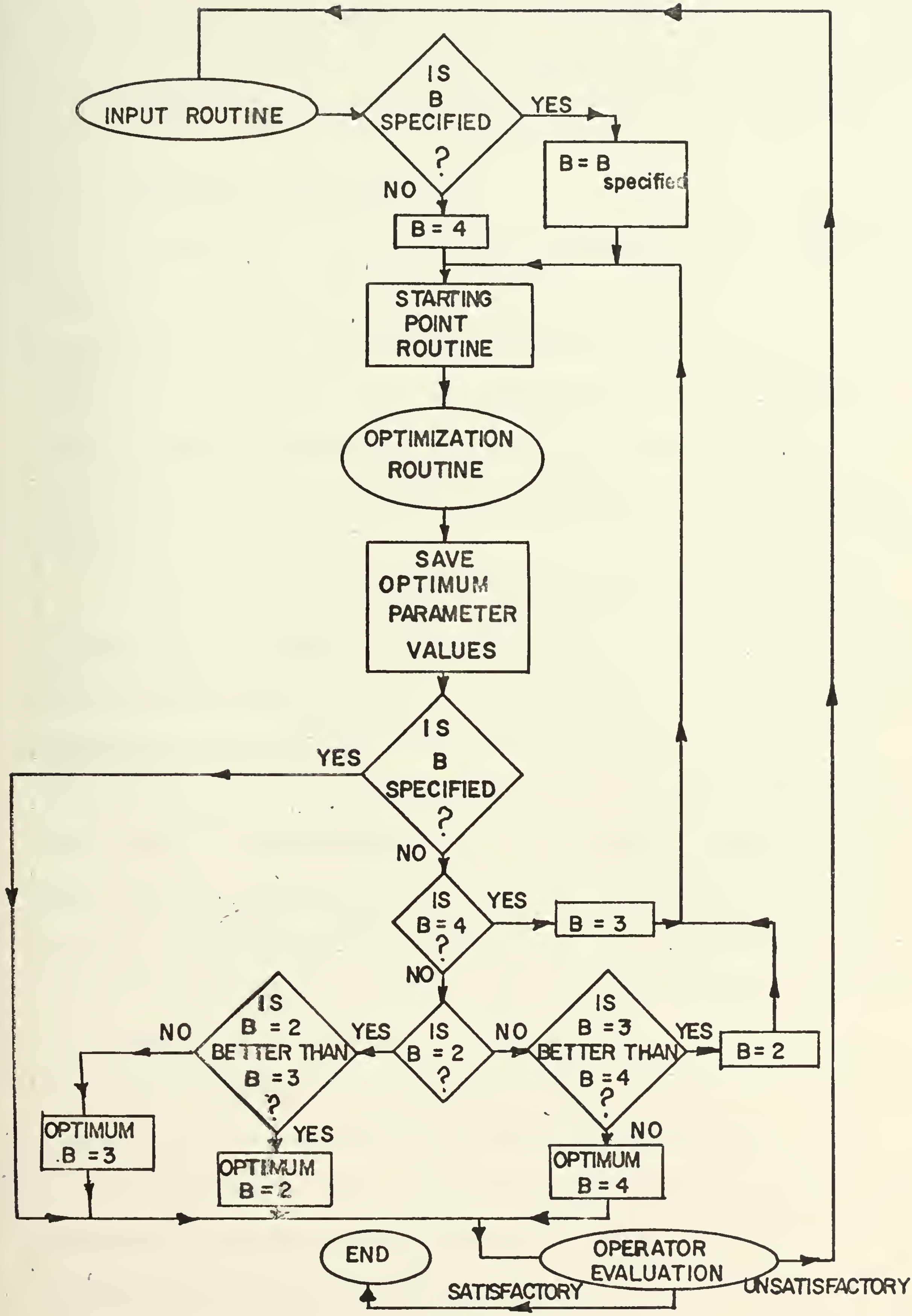

FIGURE 7
SEARCH PROCEDURE FOR NUMBER OF BLADES

search procedure ends, assuming that a 2-blade propeller would result in a still more inferior performance. However, should the 3-blade propeller outperform the 4-blade prop, then a 2-blade search is conducted.

## D. PERFORMANCE EVALUATION ROUTINE

The purpose of the Performance Evaluation Routine is to calculate aircraft performance for a given propeller configuration. This is accomplished by use of propeller performance maps found in Ref. 7. These charts give static thrust data and efficiency data for propellers of varying number of blades, activity factor, and integrated lift coefficient. These charts were reduced to data points and are fed into the program on punched cards.

According to Hamilton Standard Division of United Aircraft, the authors of Ref. 7, these data were derived semi-empirically: partially by wind tunnel data and partially by computer computation using a modified vortex theory originally developed by Goldstein in 1929. The technique employed by the PROPOP routine is to reproduce these performance charts by Lagrangian non-linear interpolation of second order. It was found that when each efficiency map was reduced to 42 data points, the accuracy with which this technique could reproduce the full chart was normally within 1%. This was considered quite adequate, as the data are assumed only to be accurate to within 1 - 2%.

Another approach to this calculation might have been to start with vortex theory and let the computer compute the efficiencies just as Hamilton Standard has done. In this way the program would not have been limited to the range of data presented in the charts. However, since data were already available covering a rather substantial range, the

added complexity of the program and the additional computer time which would be required to recalculate this information were considered too great a burden to endure for the amount of aesthetic satisfaction which might have been obtained.

Corrections for compressibility and nacelle blocking effects are presented in graphical form in supplements to Ref. 7. These charts were reduced to data points in the same manner as the efficiency charts and were incorporated into the PROPOP data bank. The accuracy of these corrections is somewhat less than the accuracy of the original data, but they are useful in determining trends. The general operation of the Evaluation Routine is outlined in Fig. 8.

Turbine engines present a problem when it is necessary to calculate their performance under a wide range of operating conditions. In a program such as this, it was considered important to reduce the amount of data required of the operator to a minimum. It was therefore necessary to make assumptions on the variation of engine performance with power setting, altitude, and velocity. The assumptions made are as follows:

1. Shaft horsepower at a given power setting is assumed to vary linearly with altitude and non-linearly (second-order) with velocity.

2. Jet thrust at a given power setting is assumed to be constant with altitude, and to vary linearly with velocity.

3. Fuel consumption at cruise power and altitude is assumed to vary linearly with velocity.

4. Constant engine RPM and constant gear ratio.

5. Takeoff, maximum speed, and climb rate are calculated at same power setting ("maximum power").

6. Range, endurance, and cruise speed are calculated at "cruise power" setting and cruise conditions.

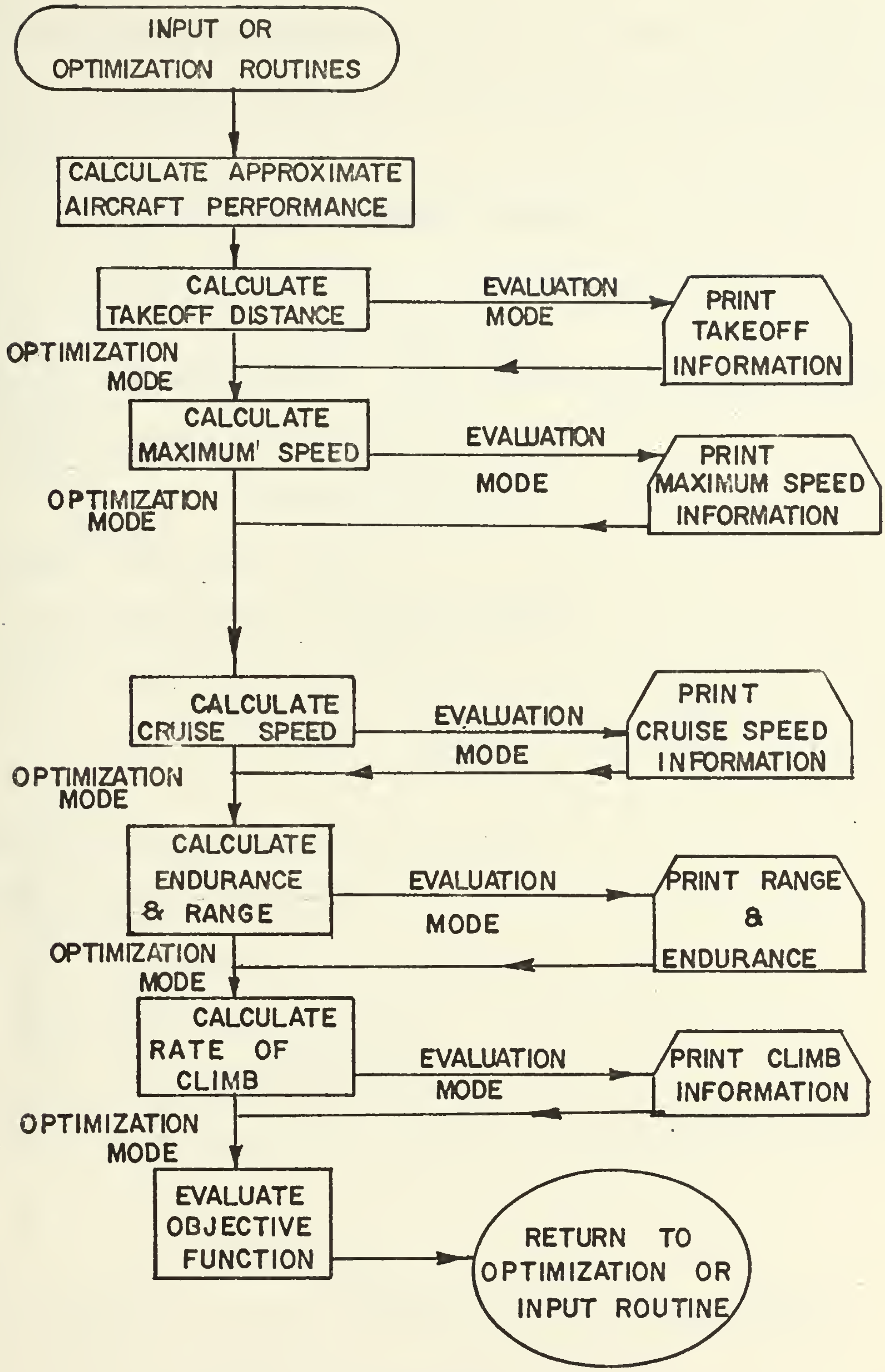
INPUT OR OPTIMIZATION ROUTINES
CALCULATE APPROXIMATE AIRCRAFT PERFORMANCE
CALCULATE TAKEOFF DISTANCE
EVALUATION MODE
PRINT TAKEOFF INFORMATION
OPTIMIZATION MODE
CALCULATE MAXIMUM SPEED
EVALUATION MODE
PRINT MAXIMUM SPEED INFORMATION
OPTIMIZATION MODE
CALCULATE CRUISE SPEED
EVALUATION MODE
PRINT CRUISE SPEED INFORMATION
OPTIMIZATION MODE
CALCULATE ENDURANCE & RANGE
EVALUATION MODE
PRINT RANGE & ENDURANCE
OPTIMIZATION MODE
CALCULATE RATE OF CLIMB
EVALUATION MODE
PRINT CLIMB INFORMATION
OPTIMIZATION MODE
EVALUATE OBJECTIVE FUNCTION
RETURN TO OPTIMIZATION OR INPUT ROUTINE

FIGURE 8

AIRCRAFT PERFORMANCE EVALUATION ROUTINE

With the above assumptions and the data furnished to the program by the operator, engine performance is calculated at all necessary operating conditions.

### 1. Calculation of Takeoff Distance

a. Assumptions

(1) $C_D = C_{D_p} + C_{D_i}$

where: $C_{D_p}$ = aircraft parasitic drag coefficient in takeoff configuration

$C_{D_i}$ = induced drag coefficient

(2) $C_{D_i} = C_L^2 / \Pi AR e$

where: $AR$ = wing aspect ratio

$e$ = wing planform efficiency factor

$C_L$ = aircraft lift coefficient

(3) Thrust $= T_{STAT} + \left[T_{LO} - T_{STAT}\right] \frac{V^2}{V_{LO}^2} = T_{STAT} + B V^2$

where: $T_{STAT}$ = static thrust (zero velocity)

$T_{LO}$ = thrust at liftoff

$V$ = instantaneous velocity (TAS)

$V_{LO}$ = liftoff velocity (TAS)

$B = \left[T_{LO} - T_{STAT}\right] / V_{LO}^2$

Thrust is assumed to vary parabolically with velocity during takeoff run.

b. Calculations

acceleration = a = F/M = summation of forces/aircraft mass

F = thrust - friction drag - aerodynamic drag

Drag = D $= \mu\left[W - L\right] + \frac{1}{2}\rho V^2 S \left[C_{D_p} + \frac{C_L^2}{\Pi AR e}\right]$

where: $\mu$ = coefficient of friction between runway and wheels

$W$ = takeoff weight of aircraft

$L$ = aircraft lift

$\rho$ = air density

$S$ = aircraft wing area

or: $$D=\mu\left[W-\tfrac{1}{2}\rho V^2 S C_L\right]+\tfrac{1}{2}\rho V^2 S\left[C_{D_p}+\frac{C_L^2}{\Pi AR e}\right]$$

In order to find the optimum $C_L$ for minimum drag:

$$\frac{dD}{dC_L}=-\tfrac{1}{2}\mu\rho V^2 S+\frac{\rho V^2 S}{\Pi AR e}\,C_{L_{opt}}=0$$

$$C_{L_{opt}}=\tfrac{1}{2}\mu\,\Pi AR e$$

Aircraft is assumed to maintain a constant $C_L$ ($C_{L_{opt}}$) during takeoff run.

$$a=\frac{g}{W}\left\{T_{STAT}+BV^2-\mu\left[W-\tfrac{1}{2}\rho V^2 S C_{L_{opt}}\right]-\tfrac{1}{2}\rho V^2 S C_D\right\}$$

or: $$a=\frac{g}{W}\left\{(T_{STAT}-\mu W)+\left[B+\tfrac{1}{2}\rho S(\mu C_{L_{opt}}-C_D)\right]V^2\right\}$$

Takeoff distance ($X_{to}$):

$$x_{to}=\int_0^{t_{LO}}V\,dt=\int_0^{t_{LO}}V\,\frac{dt}{dv}\,dV=\int_0^{V_{LO}}\frac{V}{a}\,dV$$

$$x_{to}=\frac{W}{g}\int_0^{V_{LO}}\frac{V\,dV}{A+ZV^2}$$

where: $A = T_{STAT}-\mu W$

$Z = (T_{LO}-T_{STAT})\frac{1}{V_{LO}^2}+\tfrac{1}{2}\sigma\rho_{SL}S(\mu C_{L_{opt}}-C_D)$

$\rho_{SL}$ = air density at sea level

$\sigma = \rho/\rho_{SL}$

so:

$$x_{to} = \frac{W}{2gZ} \ln\left[V^2 + \frac{A}{Z}\right]_0^{V_{LO}} = \frac{W}{2gZ} \ln\left[\frac{V_{LO}^2 + A/Z}{A/Z}\right]$$

Further assumptions:

(1) $\mu = .025$

(2) no wind

(3) no runway slope

(4) standard temperature

(5) $\sigma$ = exp (-altitude/28900)

This procedure is explained in greater detail in Ref. 8. Takeoff distance can vary greatly with pilot technique, temperature, etc., but this is of little concern to the optimization of a propeller. In general, a good takeoff propeller under the assumed conditions would also be a relatively good takeoff propeller under other conditions. Relative performance of one propeller configuration to another is the important factor.

### 2. Calculation of Endurance and Range

#### a. Endurance

Endurance is calculated at cruise speed, cruise power and altitude on the basis of a 45 minute reserve.

Endurance = (fuel capacity/fuel consumption)-45 minutes.

Fuel consumption is assumed constant.

#### b. Range

Range is also calculated at cruise speed, cruise power and altitude on the basis of a 45 minute reserve.

$$\text{Range} = \text{Endurance} \times V_{ave}$$

where: $V_{ave}$ = average velocity during flight or

$$V_{cr} + \frac{1}{2}\Delta V$$

$\Delta V$ = cruise velocity with empty fuel tanks minus cruise velocity with full fuel ($V_{cr}$)

$\Delta V$ is estimated as follows:

$$HP_{AVAIL} = \frac{TV}{550} = \frac{DV}{550} \rightarrow \quad V = \frac{HP_{AVAIL}}{D} \cdot 550$$

where:
$$D = \left[C_{D_p} + \frac{C_L^2}{\Pi ARe}\right]\frac{1}{2}\rho V^2 S$$

and
$$C_L = \frac{2W}{\rho V^2 S}$$

$$\therefore V = \frac{550\, HP_{AVAIL}}{\frac{1}{2}\rho V^2 S}\left[\frac{1}{C_{D_p} + \frac{4W^2}{\rho^2 V^4 S^2 \Pi ARe}}\right]$$

or

$$(\rho^2 S^2 \Pi ARe\, C_{D_p})V^4 - 1100\, HP_{AVAIL}\, \rho S \Pi ARe V + 4W^2 = 0$$

Taking the derivative with respect to $W$ :

$$dV = \frac{2W}{\rho S \Pi ARe}\left[\frac{dW}{275\, HP_{AVAIL} - \rho S C_{D_p} V^3}\right]$$

or approximately

$$\Delta V = \frac{2W}{\rho S \Pi ARe}\left[\frac{\Delta W / \Delta t}{275\, HP_{AVAIL} - \rho S C_{D_p} V^3}\right]\Delta t$$

where:

$$\Delta V = V_{final} - V_{initial}$$

$\Delta t$ = endurance

$-\Delta W/\Delta t$ = fuel consumption rate

$V$ = cruise velocity ($V_{cr}$)

$W = W_{GROSS} - \frac{1}{2} W_{FUEL}$ = average weight

Accounting for increased velocity due to loss of fuel weight typically increases range by 2-3%.

Range and endurance are calculated at cruise speed and conditions for several reasons. If given conditions were not specified, these performance characteristics would have to be determined by an iterative process of varying the engine power setting to determine the best setting and the corresponding aircraft velocity for best endurance. Such a process, although conceptually straight forward, would require more data than are presently required by the Input Routine. For instance, engine fuel consumption would have to be known throughout the entire range of engine operation. The iteration process would also require extensive computational time. In fact, such a procedure would probably double the time required by the entire Evaluation Routine. These penalties were considered too great to justify the computation.

### 3. Calculation of Maximum Velocity and Cruise Velocity

Calculation of maximum aircraft velocity and cruise velocity are essentially identical and the computational procedure is outlined in Fig. 9. Basically the procedure involves locating the intersection of the power available curve and the power required curve (both vs. velocity) by iterating velocity [Ref. 8].

The symbols in Fig. 9 are as follows:

$\Delta$ - step size or velocity increment

N1,N2 - indicators which serve as cues for the routine to branch in the proper direction at certain points

$HP_R$ - thrust horsepower required for level flight at given conditions

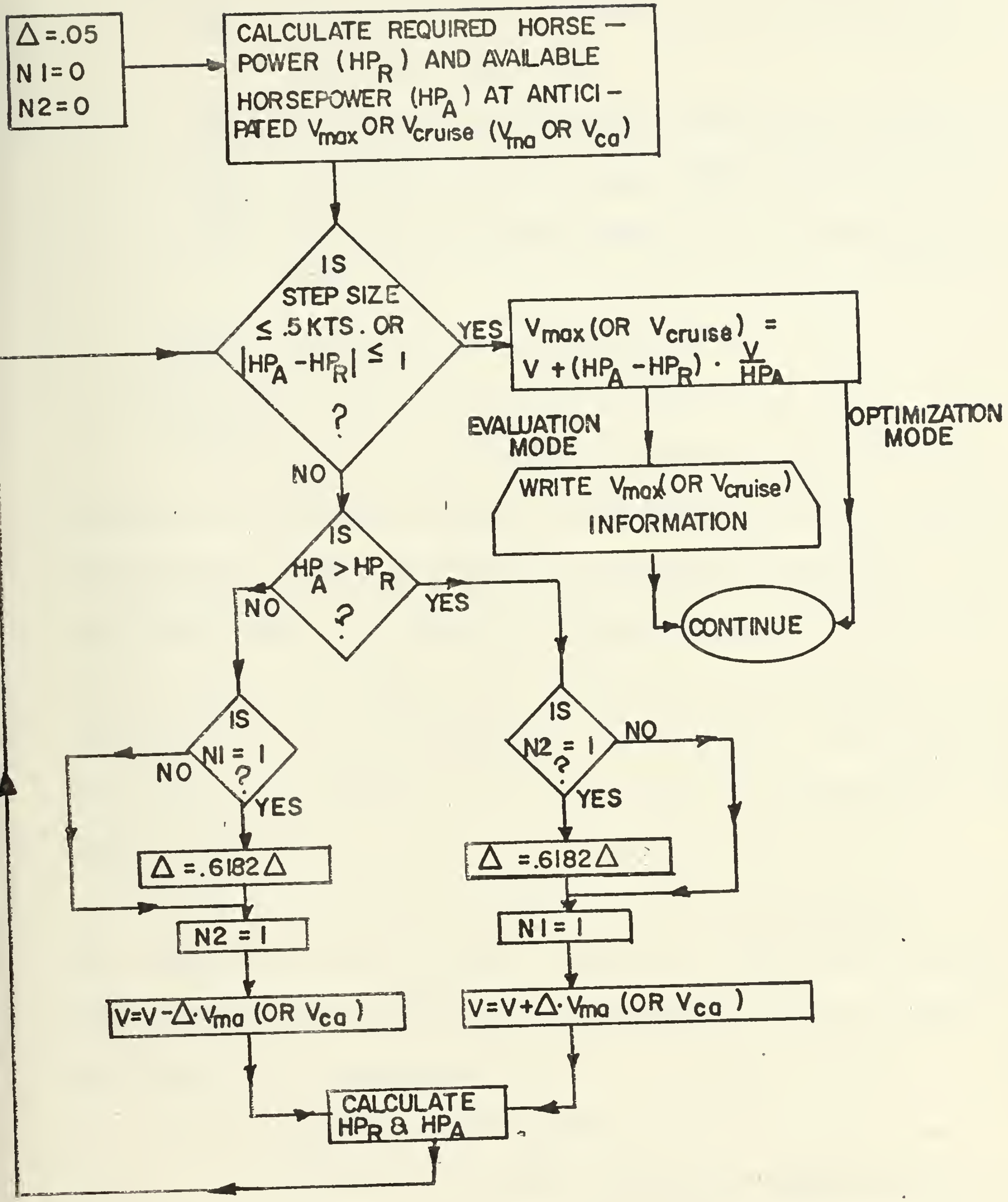
$\Delta = .05$
N1 = 0
N2 = 0
CALCULATE REQUIRED HORSE-POWER ($HP_R$) AND AVAILABLE HORSEPOWER ($HP_A$) AT ANTICIPATED $V_{max}$ OR $V_{cruise}$ ($V_{ma}$ OR $V_{ca}$)
IS STEP SIZE ≤ .5 KTS. OR $|HP_A - HP_R| \leq 1$ ?
YES
$V_{max}$ (OR $V_{cruise}$) = $V + (HP_A - HP_R) \cdot \frac{V}{HP_A}$
EVALUATION MODE
OPTIMIZATION MODE
WRITE $V_{max}$ (OR $V_{cruise}$) INFORMATION
CONTINUE
NO
IS $HP_A > HP_R$ ?
NO
YES
IS N1 = 1 ?
NO
YES
IS N2 = 1 ?
NO
YES
$\Delta = .6182\Delta$
$\Delta = .6182\Delta$
N2 = 1
N1 = 1
$V = V - \Delta \cdot V_{ma}$ (OR $V_{ca}$)
$V = V + \Delta \cdot V_{ma}$ (OR $V_{ca}$)
CALCULATE $HP_R$ & $HP_A$

FIGURE 9

CALCULATION OF MAXIMUM VELOCITY AND CRUISE VELOCITY.

velocity step size was set at .5% of the anticipated climb velocity, so this is the accuracy of the resulting climb velocity. However, at the maximum climb speed, excess thrust horsepower normally varies very slowly, so the accuracy of the actual rate of climb calculation should be much better than .5% given perfect information.

The symbols used in Fig. 10 are as follows:

$\Delta$ - step size (.005 times anticipated climb speed)

$NI$ - indicator which serves as cue for a switching point in the routine

$V$ - velocity of aircraft used in calculating excess thrust horsepower (TAS)

$\Delta HP_1, \Delta HP_2$ - excess thrust horsepower at two consecutive velocity trials

$HP_R$ - required thrust horsepower for level flight under given conditions

$HP_A$ - available thrust horsepower at given conditions

$W_{gross}$ - aircraft gross weight

R. C. - rate of climb (fpm)

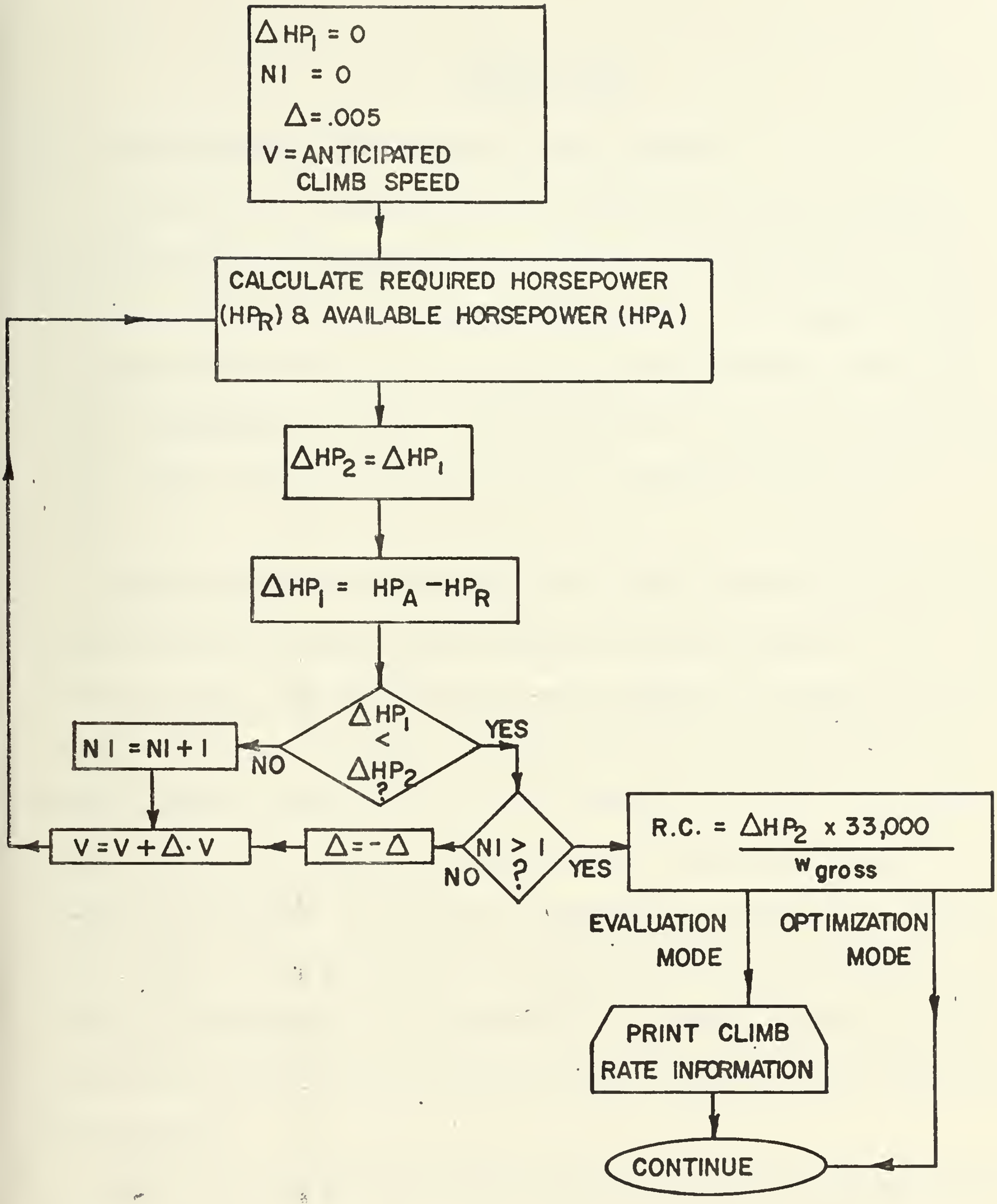
$\Delta HP_1 = 0$
$NI = 0$
$\Delta = .005$
V = ANTICIPATED CLIMB SPEED
CALCULATE REQUIRED HORSEPOWER ($HP_R$) & AVAILABLE HORSEPOWER ($HP_A$)
$\Delta HP_2 = \Delta HP_1$
$\Delta HP_1 = HP_A - HP_R$
$\Delta HP_1 < \Delta HP_2$ ?
YES
NO
$NI = NI + 1$
$V = V + \Delta \cdot V$
$\Delta = -\Delta$
$NI > 1$ ?
NO
YES
$R.C. = \frac{\Delta HP_2 \times 33{,}000}{w_{gross}}$
EVALUATION MODE
OPTIMIZATION MODE
PRINT CLIMB RATE INFORMATION
CONTINUE

FIGURE 10
CALCULATION OF CLIMB RATE

# III. PROGRAM USAGE

## A. OPTIMIZATION OF PROPELLERS FOR GIVEN AIRCRAFT

The first step in optimization of a propeller by use of PROPOP is the gathering of the required data, as listed in Section II B, on the aircraft, engines, and operational requirements. It is important that all engine horsepower (and jet thrust, if turbine powered) information used is installed engine data. Specifications as to operating conditions of the engine must be followed exactly or erroneous calculations will result.

One of the features of PROPOP is that the horsepower limit specified by the operator is automatically imposed on turbine engines. It is therefore unnecessary (and undesirable) for the operator to impose this limit on the horsepower data typed into the program (see Section II B.2.c.). PROPOP recreates engine power curves by fitting given data points with either second order curves or straight lines. These curves could be drastically altered if the operator imposed such artificial constraints on the data. It may be necessary, therefore, to extrapolate engine power curves into regions where the engines do not actually operate in order to provide the program with smoothly varying curves by which trends may be calculated.

Steps 1, 2, and 3 of the Input Routine are required for operation of PROPOP in the optimization mode. Step 4 (Aircraft Performance Constraints) may be omitted altogether. In such a case no constraints are imposed on the aircraft performance. Should the operator desire to specify some, but not all the performance constraints, he may indicate those to be left unspecified by assigning the value zero (0.) to those constraints.

Step 5 (Propeller Constraints) is required, however any constraint, except "maximum blade diameter," may be left unspecified by assigning a value of zero (0.) to that constraint. The range of data acceptable for this step is given in Section II B.5.

Step 6 (Optimization Criteria) is required in the optimization mode. In this step the operator specifies the aircraft performance characteristics for which he requires an optimum propeller. He may specify one or more of these characteristics (Section II B.6) by assigning "values of importance" from zero to ten. The highest values of importance receive the highest priority in the Optimization Routine. It has been found that the best technique for assigning these values of importance is to specify only one, giving zero (0.) as the value of the others. An attempt has been made to insure that the resulting aircraft performance closely reflects the relative values of importance assigned by the operator (in the case of several specified values of importance), however this is not always the case. The operator should specify only one characteristic, such as cruise performance, for the first trial. If the resulting propeller configuration gives other performance characteristics which the operator considers undesirable (such as large takeoff distance), then he may return to the Input Routine (Step 4) and specify a maximum allowable takeoff distance or other constraint.

After much testing of PROPOP it was found that most of the objective functions which must be optimized exhibit a "broad bottom" phenomenon. This means that in the region of the desired minimum fairly large changes in the variables may result in very small changes in the function being optimized. As a result of this situation there are many combinations of propeller configuration variables which yield nearly the same value for

Step 5 (Propeller Constraints) is required, however any constraint, except "maximum blade diameter," may be left unspecified by assigning a value of zero (0.) to that constraint. The range of data acceptable for this step is given in Section II B.5.

Step 6 (Optimization Criteria) is required in the optimization mode. In this step the operator specifies the aircraft performance characteristics for which he requires an optimum propeller. He may specify one or more of these characteristics (Section II B.6) by assigning "values of importance" from zero to ten. The highest values of importance receive the highest priority in the Optimization Routine. It has been found that the best technique for assigning these values of importance is to specify only one, giving zero (0.) as the value of the others. An attempt has been made to insure that the resulting aircraft performance closely reflects the relative values of importance assigned by the operator (in the case of several specified values of importance), however this is not always the case. The operator should specify only one characteristic, such as cruise performance, for the first trial. If the resulting propeller configuration gives other performance characteristics which the operator considers undesirable (such as large takeoff distance), then he may return to the Input Routine (Step 4) and specify a maximum allowable takeoff distance or other constraint.

After much testing of PROPOP it was found that most of the objective functions which must be optimized exhibit a "broad bottom" phenomenon. This means that in the region of the desired minimum fairly large changes in the variables may result in very small changes in the function being optimized. As a result of this situation there are many combinations of propeller configuration variables which yield nearly the same value for

a particular aircraft performance characteristic. However the performance characteristics not considered in the optimization may vary greatly with each configuration. For example, two different combinations of gear ratio, diameter, and activity factor may yield the same optimum cruise speed, but takeoff distance for one combination of these variables may be twice that for the other configuration.

Due to "noise" in the data, this broad bottom phenomenon may cause problems of apparent multimodality, or multiple optima. The operator may be surprised and somewhat disillusioned if he optimizes twice for the same performance characteristic and PROPOP gives two very different propeller configurations. However it should make no difference which configuration the operator chooses, since they both provide the same performance. This problem may be greatly reduced by specifying constraints on several performance characteristics, as mentioned above, while optimizing for a single characteristic.

The PROPOP output permits the operator to follow the program's progress through the optimization procedure. After completing the search for a particular number of blades, PROPOP gives the number of function evaluations performed. This is the number of trial solutions attempted in the search. Five hundred (500) solutions are the maximum permitted. If 500 function evaluations are performed, PROPOP gives zero (0.) for the value of this number. This is an indication to the operator that an optimum solution has not been reached, and if more accuracy is desired the limiting number of evaluations must be raised above 500. This may be done simply by changing the next to last calling argument in the "CALL DIRECT" statement, which is statement number 15, near the end of the

INPUT SUBROUTINE (Appendix A). However, such a problem is highly unlikely, as the use of more than 200 evaluations is a very rare occurrence. More information on the optimization subroutine (DIRECT) may be found in the comments preceeding that subroutine (Appendix A).

For the direct search optimization procedure (Section II C.), best results were found by starting with a rather large step size and decreasing it slowly until the desired accuracy was obtained. Consequently, the beginning normalized step size (DELCAP) was set at .816. The step length is decreased to one fourth of its current value (RHO = .25) whenever finer resolution is required. The minimum step size used is .0127. This value has proven to be sufficiently small, and further reduction rarely changes the results of the search. However, should the operator so desire, any of the above values may be changed by consulting the comment section of SUBROUTINE DIRECT and then altering the calling arguments in the "CALL DIRECT" statement previously mentioned. DELCAP is changed by altering statement 8 in SUBROUTINE INPUT. The current values of these arguments are printed in the PROPOP output before each search.

DIRECT operates by assigning the same step size to all variables. This is obviously a disadvantage when the variables have vastly different ranges, such as zero to infinity, and .15 to .7. Consequently, it was arranged that DIRECT would deal only with normalized variables rather than the actual ones. The actual variable values are therefore divided by an assigned step size before being used by DIRECT. The step sizes assigned are as follows:

diameter..........1 ft. (actual value of variable)

gear ratio.......... .5 (CSTG)

activity factor ... 90 (CSTA)

integrated $C_L$...... .1 (CSTI)

So the beginning step size for activity factor, for example, is 73.4 (or .816 x 90). In selecting these step sizes, an attempt was made to have the contours of the objective function be as nearly circular as possible near the optimum. In other words a change of one step size in any of the variables should have approximately the same effect on the propeller performance. Such a condition makes optimization more efficient. These step sizes, except that for diameter, may be altered by the operator by changing the values of CSTG, CSTA, or CSTI which are found near the beginning of PROPOP (Appendix A).

Much effort has been placed on restraining the optimization search to the limits of the data available to PROPOP. This is a relatively simple task for variables such as activity factor and integrated lift coefficient because of the well defined limits to these variables in the data. It is very much more difficult, however, for power coefficient ($C_p$) and advance ratio (J). The difficulties arise for several reasons. Power coefficient, for example, has no well defined limits in the data; the limits being different for each chart in Ref. 7. Another problem is that the search procedure does not actually fix $C_p$ or J. Instead it sets other propeller parameters and later calculates $C_p$ and J while determining aircraft performance. By that time errors may already have occurred. Because of these complexities, no attempt was made to constrain $C_p$. Advance ratio, J, is constrained rather crudely to an upper limit of three (3). The value of J is computed (V/ND) assuming a velocity which is the anticipated maximum velocity specified by the operator. If this value is greater than three, PROPOP alters either gear

ratio, to change propeller RPM (N), or diameter (D), so that J equals three. If the actual maximum velocity is greater than that assumed by the operator, however, J will still lie outside the available data. A small overshoot in $C_p$ or J is usually not fatal, as the extrapolation employed by PROPOP is usually fairly accurate close to the data limits. Large overshoots, however, can be disasterous, yielding highly inaccurate results (possibly efficiencies less than zero or greater than 100%).

If the value of J corresponding to maximum speed should be very close to three after an optimization, there is a good possibility that the propeller diameter or gear ratio have been altered during the procedure to restrain the search to the data limits. In this situation it is sometimes possible to obtain a better optimum by increasing the assumed maximum speed (and changing the appropriate corresponding engine data) at the risk of overshooting the bounds of the data.

Due to the process employed in imposing limits on propeller diameter, activity factor and integrated $C_L$, PROPOP occasionally gives an optimum propeller with one or more of the parameters very close (well within one initial step size) to its limiting value. In such a case, a slightly better optimum may sometimes be found by constraining that parameter to its limiting value. The operator should not be too surprised, however, if the results of this maneuver yield a propeller with the other variables radically changed from those of the first "optimum" propeller due to the multimodal phenomenon discussed previously in this section. In any case the aircraft performance should not vary more than one percent.

Output provided by PROPOP includes the following information on aircraft performance with the optimized propeller:

1. Takeoff
    a. Takeoff distance
    b. Static Power Coefficient
    c. Static Thrust Coefficient/Power Coefficient ($C_T/C_p$)
    d. Propeller efficiency at liftoff
    e. Blade angle (.75R) at zero velocity
    f. Power coefficient at liftoff (CP)
    g. Advance ratio at liftoff (J)
    h. Effective Mach number at liftoff (M(EFF))
    i. Compressibility correction applied to prop efficiency (FT)
    j. Blocking correction applied to prop efficiency (FB)
    k. Brake/shaft horsepower at liftoff (BHP)
    l. Jet thrust (turbine) (TJET)
2. Maximum Speed
    a. Maximum speed
    b. Blade angle (.75R)
    c. Prop efficiency
    d. Power coefficient (CP)
    e. Advance ratio (J)
    f. Effective Mach number (M(EFF))
    g. Compressibility correction applied to prop efficiency (FT)
    h. Blocking correction applied to prop efficiency (FB)
    i. Brake/shaft horsepower (BHP)
    j. Jet thrust (turbine) (TJET)
3. Cruise
    a. Cruise speed
    b. Same values as b.-j. for maximum speed

c. Range

d. Endurance

4. Climb

a. Maximum rate of climb

b. Climb speed

c. Same values as b.-j. for maximum speed

Effective Mach number is the aircraft Mach number corrected by a factor involving blade integrated $C_L$. Effective Mach number, compressibility corrections (FT), and blocking corrections (FB) are explained in supplements to Ref. 7. Listed values of prop efficiency are for corrections already applied. The blocking corrections are six percent below those given in supplements to Ref. 7 to account for interference and slipstream effects not included in the propeller efficiency data. This additional factor was suggested by Lockheed California Company [Ref. 9]. For takeoff performance, where ground effect is present, this additional blocking factor is omitted and only that factor given in supplements to Ref. 7 is applied.

## B. EVALUATION OF A GIVEN PROPELLER CONFIGURATION

As explained in Section II B., PROPOP may be operated in the evaluation mode by specifying all blade parameters in Step 5 of the Input Routine. Again, as in the optimization mode, Steps 1, 2, 3, and 5 are required. However, Steps 4 and 6 are omitted.

Output data on aircraft performance is identical with that described in part A of this section for operation in the optimization mode, except of course there is no output corresponding to the optimization procedure itself.

# IV. EXAMPLE PROBLEM: LOCKHEED P-3C

The Lockheed P-3C aircraft was chosen as the test case for the PROPOP system. Choice of this aircraft was based on its use by the Navy and availability of data on the aircraft and propellers.

## A. EVALUATION

As nearly as could be determined from information provided by the propeller and aircraft manufacturers, the P-3 propeller was optimized for a low power cruise (about 60% NRP) at an altitude of 15,000 feet, and at a speed of 325 kts. (TAS). It appears, however, that the aircraft will not fly at these conditions at full gross weight (135,000 lbs.). Therefore, a reduced weight of 105,000 pounds was used in calculations of cruise performance. Even at its empty weight of 66,211 pounds, the aircraft will not attain 325 kts. It therefore seems possible that the propellers were designed for conditions at which the aircraft cannot operate.

Data used in the P-3 performance evaluation is given in Table I. PROPOP calculates takeoff distance, maximum speed, and rate of climb at the same power setting ("maximum power"). It was therefore necessary to conduct two evaluations. Takeoff distance was calculated using takeoff power (1970°F T.I.T.), while maximum speed and climb rate were calculated at military power (1920°F T.I.T.). A third evaluation was necessary for cruise speed, range and endurance, inasmuch as a reduced gross weight was employed.

It may be noted in Table I that several of the shaft horsepower figures are considerably above the 4500 SHP gearbox limit. This necessity was discussed in Sections II B.2. and III A.

## TABLE I

## P-3C DATA SHEET

AIRCRAFT DATA:

| | |
|---|---|
| Gross Weight | 135,000 (lbs.)<br>(105,000 for cruise) |
| Wing Area | 1300 (sq. ft.) |
| Number of Engines | 4 |
| Aspect Ratio | 7.5 |
| Parasitic Drag Coeff., $C_{D_p}$ (takeoff) | .05 (10 AERO |
| " " " (cruise) | .0247 15D PYLONS) |
| Wing Efficiency Factor (e) | .98 |
| Fuel Capacity | 59,370 (lbs)<br>(29,370 for cruise) |
| Liftoff Speed | 136 @ (kts. TAS)<br>121 @ 105,000 lbs. |

ENGINE DATA:

| | | |
|---|---|---|
| Engine RPM | 13,820 | |
| SHP (assumed cruise power, altitude, and speed) | 1755 | 1450°F T.I.T. @ 25°C |
| SHP (assumed cruise power, altitude, and speed + 50 kts.) | 1861 | |
| SHP (max. power, 10,000 ft. MSL, assumed max. speed) | 4453 (T.O. @ 4637) | 0°C |
| S.L. SHP (max. pwr., 0 kts.) | 4305 (T.O. @ 4527) | |
| S.L. SHP (" " ,100 kts.) | 4486 (T.O. @ 4712) | |
| S.L. SHP (" " ,400 kts.) | 5560 (T.O. @ 5790) | |
| S.L. Jet Thrust (max. pwr., 0 kts.) | 627 (T.O. @ 639) lbs. | |
| " " " (" " ,100 kts.) | 520 (T.O. @ 533) lbs. | |
| " " " (cruise " , 0 kts.) | 462 lbs. | 1450°F T.I.T. @ 25°C |
| " " " (" " ,100 kts.) | 353 lbs. | |
| Fuel Flow (assumed cruise power, altitude, and speed) | 1035 lbs/hr | |
| Fuel Flow (assumed cruise power, altitude, and speed + 50 kts.) | 1065 lbs/hr | |
| Limiting SHP | 4500 (gearbox limit) | |

ENVIRONMENT & ANTICIPATED SPEEDS:

| | | |
|---|---|---|
| Takeoff Altitude | 0 | ft. MSL |
| Climb Altitude | 0 | " " |
| Cruise Altitude | 15,000 | " " |
| Max. Speed Altitude (assumed) | 10,000 | " " |
| Anticipated Climb Speed | 235 | kts. TAS |
| " Cruise " | 250 | " " |
| " Maximum " | 386 | " " |

Results of this performance evaluation and comparison with available flight test data are presented in Table II. Good correlation was obtained between the predicted and the flight test values. The largest error was obtained in takeoff distance (about 4%). The predicted value may be slightly conservative, but considering the many variables controlled by the pilot during takeoff, this is not thought to be a significant error.

TABLE II

COMPARISON OF FLIGHT TEST DATA AND PREDICTED PERFORMANCE OF P-3C AIRCRAFT

| PERFORMANCE | FLIGHT TEST | PREDICTION |
|---|---|---|
| Takeoff Distance | 4300 ft. | 4459 ft. |
| Maximum Speed | 386 kts. | 388 kts. |
| Cruise Speed | * | 287 kts. |
| Range | * | 1742 n. mi. |
| Endurance | * | 5.9 hrs. |
| Rate of Climb | 2010 fpm. | 2086 fpm. |

*Flight test data not available for chosen conditions.

## B. OPTIMIZATION OF P-3 PROPELLER

As an example, the P-3 propeller was optimized for the low power cruise discussed in part A of this section. It was assumed that the gear ratio (13.54:1) was determined by the engine manufacturer and was therefore constrained. The maximum allowable propeller diameter for structural clearance of the P-3C is 13.5 feet. Takeoff distance was constrained to under 2500 feet at a gross weight of 105,000 pounds.

A sample computer printout of the input sequence and the PROPOP output is found in Appendix B. The optimum propeller characteristics may be compared with the P-3 propeller [Ref. 10] in Table III. They vary only slightly, however the performance of the optimum propeller exceeds that predicted for the operational P-3 propeller in every respect. The performance comparison is given in Table IV.

TABLE III

COMPARISON OF OPERATIONAL P-3C PROPELLER AND SAMPLE OPTIMUM PROPELLER

| | P-3 | OPTIMUM |
|---|---|---|
| Number of Blades | 4 | 4 |
| Gear Ratio | 13.54:1 | 13.54:1* |
| Diameter | 13.5 ft. | 13.5 ft** |
| Activity Factor | 163 | 220*** |
| Integrated $C_L$ | .286 | .321 |

*Assumed given
**Maximum diameter for structural clearance
***Upper limit of data available

TABLE IV

PREDICTED PERFORMANCE COMPARISON BETWEEN OPERATIONAL P-3 PROPELLER AND SAMPLE OPTIMUM

| PERFORMANCE | P-3 | OPTIMUM |
|---|---|---|
| Takeoff | 4459 ft. | 3875 ft. |
| Maximum speed | 388 kts. | 389 kts. |
| Cruise speed | 287 kts. | 290 kts. |
| Range | 1742 n. mi. | 1853 n. mi.* |
| Endurance | 5.9 hrs. | 6.2 hrs.* |
| Climb | 2086 fpm. | 2202 fpm. |

*Based on 29,370 pounds of fuel. (105,000 pounds gross weight)

## V. EVALUATION OF PROJECT SUCCESS

It is felt that the original objectives as previously stated have been realized successfully. A practical, relatively easy-to-use computer program has been developed for use of empirical data in the process of optimum propeller selection. PROPOP has been shown to predict aircraft performance with acceptable accuracy and to improve on existing hardware design.

Time required from collected data to final propeller selection has been reduced from the sixteen man-hours necessary with a manual system to one half man-hours using PROPOP. A 97% reduction in required time with acceptable accuracy is considered sufficient incentive for further investigation into the feasibility of this system.

## VI. PROBLEMS ENCOUNTERED AND AREAS FOR FUTURE RESEARCH

Several problems were encountered in the development of the PROPOP system which, although apparently solved, may cause future difficulties. One such area is the constraint procedure employed during optimization, especially concerning $C_p$ and J as discussed in Section III A.

Further improvements could be made in the prediction of turbine engine performance. It appears possible that excessive data are presently required and the resulting predictions are not as accurate as might be hoped. Some simple functional relations for performance versus throttle setting, altitude, and velocity would be highly desirable.

More flexibility and a more sophisticated calculation of range and endurance would also be desirable. At present these characteristics are calculated rather crudely at cruise conditions.

The optimization procedure could also bear much more work. A system by which final aircraft performance relates more closely to assigned "values of importance" would be a significant improvement. The pattern-search algorithm upon which the optimization depends should be more thoroughly investigated to determine if another algorithm might be preferable for the propeller optimization problem.

The "broad bottom" phenomenon observed, which often leads to multiple optima, should be investigated further. The possibility exists of rescaling the variables in an attempt to alleviate this problem. Other approaches might also be considered.

The determination of the optimum relationship between the propeller variables for producing near-circular contours near the optimum is an

area which might be considered. The optimum initial step length and the best rate of reducing this length could also lead to improvements in program efficiency.

If accurate theoretical methods of predicting propeller performance could be substituted for the empirical data and interpolation procedure employed by PROPOP, there exists the possibility of much improvement. Data could be eliminated, accuracy might be improved, and available data limitations could be removed which would eliminate many of the constraint troubles. Such a substitution might even solve the multiple-optima problem.

PROPOP in its present form is really only a beginning, with vast room for improvement. For instance, there seems to be no reason why a similar method could not be developed for optimal design of an entire aircraft.

# APPENDIX A

## COMPUTER PROGRAM

```
      COMMON STATCP(28,4,8,3),STATCT(28,4,8,3),VJV(7),BETA(4,4,4,5,3),
     AFLTCPE(6,7,4,5,3),FLTETA(6,7,4,5,3),FLTCPB(4,4,4,5,3),NCP(3),CSTA,
     BWT,WAREA,NUMENG,NPROP,AR,CDPCR,CDPTO,E,FUEL,VTOKTS,NTYPE,CSTI,ATO,
     CBHPTO,RPMTO,TJETTO.BHPCL,RPMCL,TJETCL,BHPCR,RPMCR,TJETCR,BHPVM,
     DRPMVM,TJETVM,FCON,GR,ALTTO.ALTCR,ALTCL,ALTVM,AVCL,AVCR,AVM.TOCCNT,
     ECLCONT,RACONT,ENCONT,CRCONT,VMCONT,DMAX,NUMBL,D,AF,CLI,CSTG,VOITO,
     FVOICL,VOIRA,VOIEN,VOICR,VOIVM,TODIST,CTTO,RHOSL,ARC,G,NB,AFVST(8),
     GCLIV(4),NAF(3),NCLI(3).CT(9),AFVFLT(5),NJ(3),ARANGE,ETAV(27),VMLT,
     HVJVB(4),BETAV(27).RPM,BHPCR1,BHPCR2,BHPM,BHPTO1,BHPTO2,BHPTO3,NN,
     ITJETM1,TJETM2,TJETC1,TJETC2,FCON1,FCON2,BHPLIM,AENDUR,VMAX,BETAST,
     JETAVM,BETAVM,VCRKTS,ETACR.BETACR,RANGE,ENDUR,RC,VCLKTS,ETACL,MM,
     KBETACL,GRS,DS,AFS,CLIS,TOFAC,CRFAC,VMFAC,RAFAC,ENFAC,CLFAC,EFFM,
     LDELMV(4),VJVFT(5),EMACHV(4,5),FSUBTV(4,5),VJVFB(7),FSUBBV(7),TOLT,
     MEPAV(4,5),EPAALT(4,5),SHPV(5),NUMBLS,VCRLT,COORD(4,4),COORDF(4)
      NAMELIST/STAT/STATCP,STATCT/FLT/FLTCPE,FLTETA,FLTCPB,BETA
      NAMELIST/COMPV/DELMV,VJVFT,EMACHV,FSUBTV,VJV,VJVB,VJVFB,
     1FSUBBV,EPAV,EPAALT,SHPV
      WRITE(6,1098)
 1098 FORMAT(' DATA NOW BEING READ')
      READ(4,STAT)
      READ(4,FLT)
      READ(4,COMPV)
      WRITE(6,1099)
 1099 FORMAT(' DATA HAS BEEN READ')
      RHOSL=.002378
      G=32.2
      CSTA=90.
      CSTI=.1
      CSTG=.5
      CALL INPUT
    1 STOP
      END


      FUNCTION SIGMA(ALT)
      SIGMA=EXP(-ALT/28900.)
      RETURN
      END


      SUBROUTINE LIN(X1,X2,Y1,Y2,XZERO,YZERO)
C   LINEAR INTERPOLATION SUBROUTINE
C
C        X1,X2 = VALUES OF INDEPENDENT VARIABLE BRACKETING XZERO
C        Y1,Y2 = VALUES OF DEPENDENT VARIABLE CORRESPONDING TO X1,X2
C        XZERO = INDEPENDENT VARIABLE VALUE OF INTEREST
C        YZERO = VALUE OF DEPENDENT VARIABLE CORRESPONDING TO XZERO
C
```

```
      R=(XZERO-X1)/(X2-X1)
      YZERO=Y1+R*(Y2-Y1)
      RETURN
      END


      SUBROUTINE NONLIN(X1,X2,X3,Y1,Y2,Y3,XZERO,YZERO)
C NON-LINEAR INTERPOLATION (LAGRANGIAN) WITH ARBITRARY TABULAR INTERVAL
C
C          X1,X2,X3  = VALUES OF INDEPENDENT VARIABLE BRACKETING XZERO
C          Y1,Y2,Y3  = VALUES OF DEPENDENT VARIABLE CORRESPONDING TO X1,
C                      X2,X3
C             XZERO  = VALUE OF INDEPENDENT VARIABLE FOR WHICH VALUE OF
C                      DEPENDENT VARIABLE, YZERO, IS SOUGHT
C             YZERO  = VALUE OF DEPENDENT VARIABLE AT XZERO
C
      YZERO=(((XZERO-X2)*(XZERO-X3))/((X1-X2)*(X1-X3)))*Y1+(((XZERO-X1)
     1*(XZERO-X3))/((X2-X1)*(X2-X3)))*Y2+(((XZERO-X1)*(XZERO-X2))/
     2((X3-X1)*(X3-X2)))*Y3
      RETURN
      END


      SUBROUTINE ETA(CP,VJS,ETAP)
C
C   SUBROUTINE FOR CALCULATING PROPELLER EFFICIENCY AT A GIVEN ENGINE
C   POWER COEFFICIENT AND GIVEN PROPELLER ADVANCE RATIO
C
C     CP    = DIMENSIONLESS ENGINE POWER COEFFICIENT
C     VJS   = DIMENSIONLESS PROPELLER ADVANCE RATIO (V/ND)
C     ETAP  = PROPELLER EFFICIENCY
C
      DIMENSION ETAVP(9),ETAVP1(3)
      COMMON STATCP(28,4,8,3),STATCT(28,4,8,3),VJV(7),BETA(4,4,4,5,3),
     AFLTCPE(6,7,4,5,3),FLTETA(6,7,4,5,3),FLTCPB(4,4,4,5,3),NCP(3),CSTA,
     BWT,WAREA,NUMENG,NPROP,AR,CDPCR,CDPTO,E,FUEL,VTOKTS,NTYPE,CSTI,ATO,
     CBHPTO,RPMTO,TJETTO.BHPCL,RPMCL,TJETCL,BHPCR,RPMCR,TJETCR,BHPVM,
     DRPMVM,TJETVM,FCON,GR,ALTTO,ALTCR,ALTCL,ALTVM,AVCL,AVCR,AVM,TOCONT,
     ECLCONT,RACONT,ENCONT,CRCONT,VMCCNT,DMAX,NUMBL,D,AF,CLI,CSTG,VOITO,
     FVOICL,VOIRA,VOIEN,VOICR,VOIVM,TODIST,CTTO,RHOSL,ARC,G,NR,AFVST(8),
     GCLIV(4),NAF(3),NCLI(3),CT(9),AFVFLT(5),NJ(3),ARANGE.ETAV(27),VMLT,
     HVJVB(4),BETAV(27),RPM,BHPCR1,BHPCR2,BHPM,BHPTO1,BHPTO2,BHPTO3,NN,
     ITJETM1,TJETM2,TJETC1,TJETC2,FCON1,FCON2,BHPLIM,AENDUR,VMAX,BETAST,
     JETAVM,BETAVM,VCRKTS,ETACR,BETACR,RANGE,ENDUR,RC,VCLKTS,ETACL,MM,
     KBETACL,GRS,DS,AFS,CLIS,TOFAC,CRFAC,VMFAC,RAFAC,ENFAC,CLFAC,FFFM,
     LDELMV(4),VJVFT(5),EMACHV(4,5),FSUBTV(4,5),VJVFP(7),FSUBBV(7),TOLT,
     MEPAV(4,5),EPAALT(4,5),SHPV(5),NUMBLS,VCRLT,COORD(4,4),COORDF(4)
       DO 1 I=1,6
```

```
      IF(VJS.LT.VJV(1))GO TO 2                                          PR000930
      IF(VJS.GE.VJV(I).AND.VJS.LE.VJV(I+1))GO TO 2                      PR000940
    1 CONTINUE                                                          PR000950
    2 IF(I.EQ.6)GO TO 3                                                 PR000960
      NJ(1)=I                                                           PR000970
      NJ(2)=I+1                                                         PR000980
      NJ(3)=I+2                                                         PR000990
      GO TO 4                                                           PR001000
    3 NJ(1)=I+1                                                         PR001010
      NJ(2)=I                                                           PR001020
      NJ(3)=I-1                                                         PR001030
    4 DO 5 I=1,3                                                        PR001040
      DO 6 J=1,3                                                        PR001050
      DO 7 K=1,3                                                        PR001060
      N1=K+3*(J-1)+9*(I-1)                                              PR001070
      DO 8 N=1,5                                                        PR001080
      IF(CP.LT.FLTCPE(1,NJ(K),NCLI(J),NAF(I),NB))GO TO 9                PR001090
      IF(CP.GE.FLTCPE(N,NJ(K),NCLI(J),NAF(I),NB).AND.CP.LE.FLTCPE(N+1,  PR001100
     1NJ(K),NCLI(J),NAF(I),NB))GO TO 9                                  PR001110
    8 CONTINUE                                                          PR001120
    9 IF(N.EQ.5)GO TO 10                                                PR001130
      NCP(1)=N                                                          PR001140
      NCP(2)=N+1                                                        PR001150
      NCP(3)=N+2                                                        PR001160
      GO TO 11                                                          PR001170
   10 NCP(1)=N+1                                                        PR001180
      NCP(2)=N                                                          PR001190
      NCP(3)=N-1                                                        PR001200
   11 CALL NONLIN(FLTCPE(NCP(1),NJ(K),NCLI(J),NAF(I),NB),FLTCPE(NCP(2), PR001210
     1NJ(K),NCLI(J),NAF(I),NB),FLTCPE(NCP(3),NJ(K),NCLI(J),NAF(I),NB),  PR001220
     2FLTETA(NCP(1),NJ(K),NCLI(J),NAF(I),NB),FLTETA(NCP(2),NJ(K),NCLI(J)PR001230
     3,NAF(I),NB),FLTETA(NCP(3),NJ(K),NCLI(J),NAF(I),NB),CP,ETAV(N1))   PR001240
    7 CONTINUE                                                          PR001250
    6 CONTINUE                                                          PR001260
    5 CONTINUE                                                          PR001270
      N=1                                                               PR001280
      DO 12 I=1,25,3                                                    PR001290
      CALL NONLIN(VJV(NJ(1)),VJV(NJ(2)),VJV(NJ(3)),ETAV(I),ETAV(I+1),   PR001300
     1ETAV(I+2),VJS,ETAVP(N))                                           PR001310
      N=N+1                                                             PR001320
   12 CONTINUE                                                          PR001330
      N=1                                                               PR001340
      CLI=CLI*CSTI                                                      PR001350
      AF=AF*CSTA                                                        PR001360
      DO 13 I=1,7,3                                                     PR001370
      CALL NONLIN(CLIV(NCLI(1)),CLIV(NCLI(2)),CLIV(NCLI(3)),ETAVP(I),   PR001380
     1ETAVP(I+1),ETAVP(I+2),CLI,ETAVP1(N))                              PR001390
      N=N+1                                                             PR001400
```

```
   13 CONTINUE
      CALL NONLIN(AFVFLT(NAF(1)),AFVFLT(NAF(2)),AFVFLT(NAF(3)),ETAVP1(1)
     1,ETAVP1(2),ETAVP1(3),AF,ETAP)
      CLI=CLI/CSTI
      AF=AF/CSTA
      RETURN
      END

      SUBROUTINE BLDANG(CP,VJS,BETAP)
C
C  SUBROUTINE FOR CALCULATING PROPELLER BLADE ANGLE (.75R) AT A GIVEN
C  ENGINE POWER COEFFICIENT AND GIVEN PROPELLER ADVANCE RATIO
C
C     CP    = DIMENSIONLESS ENGINE POWER COEFFICIENT
C     VJS   = DIMENSIONLESS PROPELLER ADVANCE RATIO (V/ND)
C     BETAP = PROPELLER BLADE ANGLE (.75R)
C
      DIMENSION BETAVP(9),BETAV1(3)
      COMMON STATCP(28,4,8,3),STATCT(28,4,8,3),VJV(7),BETA(4,4,4,5,3),
     AFLTCPE(6,7,4,5,3),FLTETA(6,7,4,5,3),FLTCPB(4,4,4,5,3),NCP(3),CSTA,
     BWT,WAREA,NUMENG,NPROP,AR,CDPCR,CDPTO,E,FUEL,VTCKTS,NTYPE,CSTI,ATO,
     CBHPTO,RPMTO,TJETTO.BHPCL.RPMCL,TJETCL,BHPCR,RPMCR,TJETCR,BHPVM,
     DRPMVM,TJETVM,FCON,GR,ALTTO,ALTCR,ALTCL,ALTVM,AVCL,AVCR,AVM,TOCONT,
     ECLCONT,RACONT,ENCONT,CRCONT,VMCONT,DMAX,NUMBL,D,AF,CLI,CSTG,VOITO,
     FVOICL,VOIRA,VOIEN,VOICR,VOIVM,TCDIST,CTTO,RHOSL,ARC,G,NB,AFVST(8),
     GCLIV(4),NAF(3),NCLI(3),CT(9),AFVFLT(5),NJ(3),ARANGE,ETAV(27),VMLT,
     HVJVB(4),BETAV(27),RPM,BHPCR1,BHPCR2,BHPM,BHPTO1,BHPTO2,BHPTO3,NN,
     ITJETM1,TJETM2,TJETC1,TJETC2,FCON1,FCON2,BHPLIM,AENDUR,VMAX,BETAST,
     JETAVM,BETAVM,VCRKTS,ETACR,BETACR,RANGE,ENDUR,RC,VCLKTS,ETACL,MM,
     KBETACL,GRS,DS,AFS,CLIS,TOFAC,CRFAC,VMFAC,RAFAC,ENFAC,CLFAC,EFFM,
     LDELMV(4),VJVFT(5),EMACHV(4,5),FSUBTV(4,5),VJVFB(7),FSUBBV(7),TOLT,
     MEPAV(4,5),EPAALT(4,5),SHPV(5),NUMBLS,VCRLT,COORD(4,4),COORDF(4)
      DO 1 I=1,3
      IF(VJS.LT.VJVB(1))GO TO 2
      IF(VJS.GE.VJVB(I).AND.VJS.LE.VJVB(I+1))GO TO 2
    1 CONTINUE
    2 IF(I.EQ.3)GO TO 3
      NJ(1)=I
      NJ(2)=I+1
      NJ(3)=I+2
      GO TO 4
    3 NJ(1)=I+1
      NJ(2)=I
      NJ(3)=I-1
    4 DO 5 I=1,3
      DO 6 J=1,3
      DO 7 K=1,3
```

```
      N1=K+3*(J-1)+9*(I-1)
      DO 8 N=1,3
      IF(CP.LT.FLTCPB(1,NJ(K),NCLI(J),NAF(I),NB))GO TO 9
      IF(CP.GE.FLTCPB(N,NJ(K),NCLI(J),NAF(I),NB).AND.CP.LE.FLTCPB(N+1,
     1NJ(K),NCLI(J),NAF(I),NB))GO TO 9
    8 CONTINUE
    9 IF(N.EQ.3)GO TO 10
      NCP(1)=N
      NCP(2)=N+1
      NCP(3)=N+2
      GO TO 11
   10 NCP(1)=N+1
      NCP(2)=N
      NCP(3)=N-1
   11 CALL NONLIN(FLTCPB(NCP(1),NJ(K),NCLI(J),NAF(I),NB),FLTCPB(NCP(2),
     1NJ(K),NCLI(J),NAF(I),NB),FLTCPB(NCP(3),NJ(K),NCLI(J),NAF(I),NB),
     2  BETA(NCP(1),NJ(K),NCLI(J),NAF(I),NB),   BETA(NCP(2),NJ(K),NCLI(J)
     3,NAF(I),NB),   BETA(NCP(3),NJ(K),NCLI(J),NAF(I),NB),CP,BETAV(N1))
    7 CONTINUE
    6 CONTINUE
    5 CONTINUE
      N=1
      DO 12 I=1,25,3
      CALL NONLIN(VJVB(NJ(1)),VJVB(NJ(2)),VJVB(NJ(3)),BETAV(I),BETAV(I+
     1),BETAV(I+2),VJS,BETAVP(N))
      N=N+1
   12 CONTINUE
      N=1
      CLI=CLI*CSTI
      AF=AF*CSTA
      DO 13 I=1,7,3
      CALL NONLIN(CLIV(NCLI(1)),CLIV(NCLI(2)),CLIV(NCLI(3)),BETAVP(I),
     1BETAVP(I+1),BETAVP(I+2),CLI,BETAV1(N))
      N=N+1
   13 CONTINUE
      CALL NONLIN(AFVFLT(NAF(1)),AFVFLT(NAF(2)),AFVFLT(NAF(3)),BETAV1(1)
     1,BETAV1(2),BETAV1(3),AF,BETAP)
      CLI=CLI/CSTI
      AF=AF/CSTA
      RETURN
      END
C     ----------------------------------------------------------------
C
C        SUBROUTINE DIRECT
C
C        PURPOSE
C           TO LOCATE A LOCAL MINIMUM OF A FUNCTION, S, OF K VARIABLES
C           BY THE METHOD OF DIRECT SEARCH (HOOKE AND JEEVES)
```

```
C                                                                       PR002370
C     USAGE                                                             PR002380
C        CALL DIRECT(PSI,K,SPSI,DELCAP,RHO,DELLC,S,KONVRG,MAXEV,KN)     PR002390
C                                                                       PR002400
C     DESCRIPTION OF PARAMETERS                                         PR002410
C      PSI    - VECTOR OF K INDEPENDENT VARIABLES. IT IS FILLED INI-    PR002420
C               TIALLY BY USER WITH FIRST ESTIMATE OF SOLUTION. AT      PR002430
C               EXIT FROM DIRECT IT CONTAINS BEST VALUES ATTAINED.      PR002440
C      K      - NUMBER OF INDEPENDENT VARIABLES OF FUNCTION, S, TO      PR002450
C               BE MINIMIZED                                            PR002460
C      SPSI   - AT EXIT FROM DIRECT CONTAINS SMALLEST S(PSI) ATTAINED   PR002470
C      DELCAP - INITIAL STEP LENGTH (SAME FOR ALL VARIABLES)            PR002480
C      N.B. DELCAP IS ALTERED BY DIRECT. DO NOT USE NUMERICAL VALUE INPR002490
C      CALLING LIST.                                                    PR002500
C      RHO    - STEP REDUCTION FACTOR (0.LT.RHO.LT.1). THE VALUE .125   PR002510
C               OR .25 IS SUGGESTED                                     PR002520
C      DELLC  - END CRITERION. WHEN CURRENT STEP SIZE IS LESS THAN      PR002530
C               DELLC, THE SEARCH IS TERMINATED.                        PR002540
C      S      - THE NAME OF AN EXTERNAL FUNCTION, S(PHI), TO BE MINI-   PR002550
C               MIZED. A FUNCTION SUBPROGRAM OF THE NAME MUST BE SUP-   PR002560
C               PLIED BY THE USER.                                      PR002570
C      KONVRG - AN INDICATOR TO BE TESTED BY USER UPON EXIT FROM        PR002580
C               DIRECT. IF KONVERG                                      PR002590
C                =-1, A PARAMETER ERROR WAS DETECTED, FOR EXAMPLE       PR002600
C                     K.GT.15   OR  K.LE.0,                             PR002610
C                     DELCAP.LE.0,                                      PR002620
C                     RHO.LE.0  OR  RHO.GE.1,                           PR002630
C                     DELLC.LE.0. (RETURN IS MADE IMMEDIATELY.)         PR002640
C                =0, MAXEV WAS EXCEEDED. (SPSI IS NOT MINIMUM.)         PR002650
C                .GT.0, KONVRG = NUMBER OF EVALUATIONS OF S MADE TO     PR002660
C                    ATTAIN SPSI. (SUCCESS IS INDICATED.)               PR002670
C      MAXEV  - MAXIMUM NUMBER OF FUNCTION EVALUATIONS USER WILL ALLOWPR002680
C               IN FINDING MINIMUM. IF MAXEV.LE.0, AN EFFECTIVE VALUE   PR002690
C               OF 500 WILL BE USED.                                    PR002700
C      KN     - AN INDICATOR TO BE USED TO OBTAIN DIAGNOSTIC OUTPUT IFPR002710
C               NECESSARY. IF KN                                        PR002720
C                .LE.-1, OUTPUT OF FUNCTION VALUE AND CORRESPONDING     PR002730
C                  VARIABLES IS MADE AT THE ORIGIN, AFTER EACH EXPLORE  PR002740
C                  MOVE, AFTER EACH PATTERN (INCLUDING POST-PATTERN     PR002750
C                   EXPLORATION)  AND AT EXIT, WITH FINAL VALUE OF      PR002760
C                   KONVRG.                                             PR002770
C                =0, NO OUTPUT IS ACCOMPLISHED BY DIRECT.               PR002780
C                .GE.1, OUTPUT IS MADE AS IN THE CASE KN.LE.-1, EXCEPTPR002790
C                    EXPLORE MOVES ARE OMITTED.                         PR002800
C               NOTE.. IN CASES KN.NE.0, OUTPUT OF PARAMETERS IS ALSO   PR002810
C               MADE BEFORE SEARCH BEGINS. ALL OUTPUT IS ON MEDIUM 6.   PR002820
C                                                                       PR002830
C     METHOD                                                            PR002840
```

```
C      (1) HOOKE,R. AND JEEVES,T.A., ,,DIRECT SEARCH SOLUTION OF NUMER-PRO02850
C          ICAL AND STATISTICAL PROBLEMS,, JOURNAL A.C.M. 8(1961), P.   PRO02860
C          212-219.                                                       PRO02870
C      (2)BELL,M. AND PILE,M.C., ,,REMARK ON ALGORITHM 178, DIRECT       PRO02880
C          SEARCH,, COMM. A.C.M., 9(SEPT 1966), P.684-5.                 PRO02890
C                                                                         PRO02900
C      NOTE.. THE MODIFICATIONS PROPOSED IN REF.(2) HAVE BEEN ADOPTED   PRO02910
C          IN THIS SUBROUTINE. IN ADDITION PROCEDURE E HAS BEEN MADE    PRO02920
C          MORE EFFICIENT AND A THIRD VALUE OF KONVRG=,,CONVERGE,,      PRO02930
C          HAS BEEN USED TO INDICATE INCORRECT PARAMETER VALUES.        PRO02940
C     PROGRAMMER AND DATE                                                 PRO02950
C                                                                         PRO02960
C       R. R. HILLEARY, NOVEMBER 1966                                     PRO02970
C       MODIFIED BY R. L. SHAW, JUNE 1970                                 PRO02980
C                                                                         PRO02990
C-----------------------------------------------------------------------PRO03000
C                                                                         PRO03010
      SUBROUTINE DIRECT (PSI,K,SPSI,DELCAP,RHO,DELLC,S,KONVRG,MAXEV ,KN)PRO03020
      COMMON STATCP(28,4,8,3),STATCT(28,4,8,3),VJV(7),BETA(4,4,4,5,3),
     AFLTCPE(6,7,4,5,3),FLTETA(6,7,4,5,3),FLTCPB(4,4,4,5,3),NCP(3),CSTA,
     BWT,WAREA,NUMENG,NPROP,AR,CDPCR,CDPTO,E,FUEL,VTCKTS,NTYPE,CSTI,ATO,
     CBHPTO,RPMTO,TJETTO.BHPCL,RPMCL,TJETCL,BHPCR,RPMCR,TJETCR,BHPVM,
     DRPMVM,TJETVM,FCON,GR,ALTTO,ALTCR,ALTCL,ALTVM,AVCL,AVCR,AVM,TOCONT,
     ECLCONT,RACCNT,ENCONT,CRCONT,VMCONT,DMAX,NUMBL,D,AF,CLI,CSTG,VOITO,
     FVOICL,VOIRA,VOIEN,VOICR,VOIVM,TCDIST,CTTO,RHOSL,ARC,G,NB,AFVST(8),
     GCLIV(4),NAF(3),NCLI(3),CT(9),AFVFLT(5),NJ(3),ARANGE,ETAV(27),VMLT,
     HVJVB(4),BETAV(27),RPM,BHPCR1,BHPCR2,BHPM,BHPTO1,BHPTO2,BHPTO3,NN,
     ITJETM1,TJETM2,TJETC1,TJETC2,FCON1,FCON2,BHPLIM,AENDUR,VMAX,BETAST,
     JETAVM,BETAVM,VCRKTS,ETACR,BETACR,RANGE,ENDUR,RC,VCLKTS,ETACL,MM,
     KBETACL,GRS,DS,AFS,CLIS,TOFAC,CRFAC,VMFAC,RAFAC,ENFAC,CLFAC,EFFM,
     LDELMV(4),VJVFT(5),EMACHV(4,5),FSUBTV(4,5),VJVFB(7),FSUBFV(7),TOLT,
     MEPAV(4,5),EPAALT(4,5),SHPV(5),NUMBLS,VCRLT,COORD(4,4),COORDF(4)    PRO03190
      DIMENSION PSI(K),PHI(15),SLC(15)                                  PRO03200
      INTEGER EVAL                                                       PRO03210
C                                                                         PRO03220
      IF(K.GT.15) GO TO 50                                               PRO03230
      IF(K) 50,50,4                                                      PRO03240
    4 IF(DELCAP) 50,50,5                                                 PRO03250
    5 IF(RHO) 50,50,6                                                    PRO03260
    6 IF(RHO.GE.1.) GO TO 50                                             PRO03270
      IF(DELLC) 50,50,7                                                  PRO03280
    7 MAXEVL = MAXEV                                                     PRO03290
      IF(MAXEVL) 8,3,9                                                   PRO03300
    8 MAXEVL = 500                                                       PRO03310
C                                                                         PRO03320
    9 DO 60 I=1,K
```

```
   60 SLC(I) = DELCAP                                                   PRO03330
      IF((PSI(1)/PSI(2)).GT.((3.*RPMVM)/(101.4*AVM*CSTG)).AND.GRS.EQ.0.)PRO03340
     1PSI(1)=(3.*PSI(2)*RPMVM)/(101.4*AVM*CSTG)                          PRO03350
      IF((PSI(1)/PSI(2)).GT.((3.*RPMVM)/(101.4*AVM*CSTG)).AND.DS.EQ.0.  PRO03360
     1.AND.GRS.NE.0.)PSI(2)=(PSI(1)*101.4*AVM*CSTG)/(3.*RPMVM)           PRO03370
      IF(PSI(2).GT.DMAX)PSI(2)=DMAX                                     PRO03380
      SPSI = S(PSI)                                                     PRO03390
      EVAL = 1                                                          PRO03400
C                                                                       PRO03410
      IF(KN) 61,1,61                                                    PRO03420
   61 WRITE(6,208)NUMBL                                                 PRO03430
  208 FORMAT(/' BEGIN'I2,' BLADE SEARCH')                               PRO03440
      WRITE (6,63) DELCAP,RHO,DELLC,MAXEVL,KN                           PRO03450
   63 FORMAT (/14H1DIRECT SEARCH,2X,8HDELCAP =,E15.6,2X,5HRHO =,E15.6,  PRO03460
     12X,7HDELLC =,E15.6,2X,8HMAXEVL =,I8,2X,5H KN =,I3//8H0 MOVE  ,    PRO03470
     215H FUNCTION VALUE,7X,'GEAR RATIO',8X,'PROP DIAM.',5X,            PRO03480
     3'ACTIVITY FACTOR',4X,'INTEG. LIFT COEF.')                         PRO03490
      PSI(1)=PSI(1)*CSTG                                                PRO03500
      PSI(3)=PSI(3)*CSTA                                                PRO03510
      PSI(4)=PSI(4)*CSTI                                                PRO03520
      WRITE ( 6,62) SPSI, (PSI(I),I=1,K)                                PRO03530
   62 FORMAT(8H0ORIGIN ,E15.7,8X,F6.2,12X,F6.2,12X,F6.1,14X,F6.3)       PRO03540
      PSI(1)=PSI(1)/CSTG                                                PRO03550
      PSI(3)=PSI(3)/CSTA                                                PRO03560
      PSI(4)=PSI(4)/CSTI                                                PRO03570
C                                                                       PRO03580
    1 SS = SPSI                                                         PRO03590
      DO 10 I=1,K                                                       PRO03600
   10 PHI(I)= PSI(I)                                                    PRO03610
      ASSIGN 11 TO IBK                                                  PRO03620
      GO TO 40                                                          PRO03630
C                                                                       PRO03640
   11 IF(KN) 12,13,13                                                   PRO03650
   12 PHI(1)=PHI(1)*CSTG                                                PRO03660
      PHI(3)=PHI(3)*CSTA                                                PRO03670
      PHI(4)=PHI(4)*CSTI                                                PRO03680
      WRITE ( 6,14) SS,(PHI(I),I=1,K)                                   PRO03690
   14 FORMAT(8H0EXPLORE,E15.7,8X,F6.2,12X,F6.2,12X,F6.1,14X,F6.3)       PRO03700
      PHI(1)=PHI(1)/CSTG                                                PRO03710
      PHI(3)=PHI(3)/CSTA                                                PRO03720
      PHI(4)=PHI(4)/CSTI                                                PRO03730
C                                                                       PRO03740
   13 IF(SS.GE.SPSI) GO TO 3                                            PRO03750
    2 IF (EVAL.GE.MAXEVL) GO TO 51                                      PRO03760
C                                                                       PRO03770
      DO 20 I=1,K                                                       PRO03780
      IF(I.EQ.1.AND.GRS.NE.0.)GO TO 20                                  PRO03790
      IF(I.EQ.2.AND.DS.NE.0.)GO TO 20                                   PRO03800
```

```
      IF(I.EQ.3.AND.AFS.NE.0.)GO TO 20
      IF(I.EQ.4.AND.CLIS.NE.0.)GO TO 20
      IF(SLC(I)) 21,50,22
   21 IF(PHI(I).GT.PSI(I))  SLC(I) =-SLC(I)
      GO TO 23
   22 IF(PHI(I).LT.PSI(I))  SLC(I) = -SLC(I)
   23 THET = PSI(I)
      PSI(I) = PHI(I)
      PHI(I) = 2.*PHI(I) - THET
      IF((PHI(3).GT.(220./CSTA)).OR.(NUMBL.EQ.2.AND.PHI(3).GT.(120./CSTA
     1)))GO TO 56
      GO TO 57
   56 IF(NUMBL.EQ.2)PHI(3)=120./CSTA
      IF(NUMBL.NE.2)PHI(3)=220./CSTA
   57 IF(PHI(3).LT.(80./CSTA))PHI(3)=80./CSTA
      IF(PHI(4).GT.(.7/CSTI))PHI(4)=.7/CSTI
      IF((PHI(4).LT.(.15/CSTI)).OR.(NUMBL.EQ.2.AND.PHI(4).LT.(.3/CSTI)))
     1GO TO 58
      GO TO 59
   58 IF(NUMBL.EQ.2)PHI(4)=.3/CSTI
      IF(NUMBL.NE.2)PHI(4)=.15/CSTI
   59 IF((PHI(1)/PHI(2)).GT.((3.*RPMVM)/(101.4*AVM*CSTG)).AND.GRS.EQ.0.)
     1PHI(1)=(3.*PHI(2)*RPMVM)/(101.4*AVM*CSTG)
      IF((PHI(1)/PHI(2)).GT.((3.*RPMVM)/(101.4*AVM*CSTG)).AND.DS.EQ.0.
     1.AND.GRS.NE.0.)PHI(2)=(PHI(1)*101.4*AVM*CSTG)/(3.*RPMVM)
      IF(PHI(2).GT.DMAX.AND.DS.EQ.0.)PHI(2)=DMAX
   20 CONTINUE
C
      SPSI = SS
      SPHI=S(PHI)
      SS=SPHI
      EVAL = EVAL +1
      ASSIGN 25 TO IBK
C
   40 DO 41 I=1,K
      IF(I.EQ.1.AND.GRS.NE.0.)GO TO 41
      IF(I.EQ.2.AND.DS.NE.0.)GO TO 41
      IF(I.EQ.3.AND.AFS.NE.0.)GO TO 41
      IF(I.EQ.4.AND.CLIS.NE.0.)GO TO 41
      THET = PHI(I)
      SLCI = SLC(I)
      PHI(I)=THET+SLCI
      IF((PHI(3).GT.(220./CSTA)).OR.(NUMBL.EQ.2.AND.PHI(3).GT.(120./CSTA
     1)))GO TO 70
      GO TO 71
   70 IF(NUMBL.EQ.2)PHI(3)=120./CSTA
      IF(NUMBL.NE.2)PHI(3)=220./CSTA
   71 IF(PHI(3).LT.(80./CSTA))PHI(3)=80./CSTA
```

```
      IF(PHI(4).GT.(.7/CSTI))PHI(4)=.7/CSTI                             PRC04290
      IF((PHI(4).LT.(.15/CSTI)).OR.(NUMBL.EQ.2.AND.PHI(4).LT.(.3/CSTI)))PRC04300
     1GO TO 72                                                          PRC04310
      GO TO 73                                                          PRC04320
   72 IF(NUMBL.EQ.2)PHI(4)=.3/CSTI                                      PRC04330
      IF(NUMBL.NE.2)PHI(4)=.15/CSTI                                     PRC04340
   73 IF((PHI(1)/PHI(2)).GT.((3.*RPMVM)/(101.4*AVM*CSTG)).AND.GRS.EQ.0.)PRC04350
     1PHI(1)=(3.*PHI(2)*RPMVM)/(101.4*AVM*CSTG)                         PRC04360
      IF((PHI(1)/PHI(2)).GT.((3.*RPMVM)/(101.4*AVM*CSTG)).AND.DS.EQ.0.  PRC04370
     1.AND.GRS.NE.0.)PHI(2)=(PHI(1)*101.4*AVM*CSTG)/(3.*RPMVM)          PRC04380
      IF(PHI(2).GT.DMAX.AND.DS.EQ.0.)PHI(2)=DMAX                        PRC04390
      SPHI =S(PHI)                                                      PRC04400
      EVAL = EVAL +1                                                    PRC04410
      IF(SS.GT.SPHI)GO TO 42                                            PRC04420
      PHI(I) = THET - SLCI                                              PRC04430
      IF((PHI(3).GT.(220./CSTA)).OR.(NUMBL.EQ.2.AND.PHI(3).GT.(120./CSTAPRC04440
     1)))GO TO 75                                                       PRC04450
      GO TO 76                                                          PRC04460
   75 IF(NUMBL.EQ.2)PHI(3)=120./CSTA                                    PRC04470
      IF(NUMBL.NE.2)PHI(3)=220./CSTA                                    PRC04480
   76 IF(PHI(3).LT.(80./CSTA))PHI(3)=80./CSTA                           PRC04490
      IF(PHI(4).GT.(.7/CSTI))PHI(4)=.7/CSTI                             PRC04500
      IF((PHI(4).LT.(.15/CSTI)).OR.(NUMBL.EQ.2.AND.PHI(4).LT.(.3/CSTI)))PRC04510
     1GO TO 77                                                          PRC04520
      GO TO 78                                                          PRC04530
   77 IF(NUMBL.EQ.2)PHI(4)=.3/CSTI                                      PRC04540
      IF(NUMBL.NE.2)PHI(4)=.15/CSTI                                     PRC04550
   78 IF((PHI(1)/PHI(2)).GT.((3.*RPMVM)/(101.4*AVM*CSTG)).AND.GRS.EQ.0.)PRC04560
     1PHI(1)=(3.*PHI(2)*RPMVM)/(101.4*AVM*CSTG)                         PRC04570
      IF((PHI(1)/PHI(2)).GT.((3.*RPMVM)/(101.4*AVM*CSTG)).AND.DS.EQ.0.  PRC04580
     1.AND.GRS.NE.0.)PHI(2)=(PHI(1)*101.4*AVM*CSTG)/(3.*RPMVM)          PRC04590
      IF(PHI(2).GT.DMAX.AND.DS.EQ.0.)PHI(2)=DMAX                        PRC04600
      SPHI=S(PHI)                                                       PRC04610
      EVAL = EVAL +1                                                    PRC04620
      IF(SPHI.GE.SS) GO TO 44                                           PRC04630
      SLC(I)=-SLCI                                                      PRC04640
   42 SS=SPHI                                                           PRC04650
      GO TO 41                                                          PRC04660
   44 PHI(I)=THET                                                       PRC04670
   41 CONTINUE                                                          PRC04680
C                                                                       PRC04690
      GO TO IBK,(11,25)                                                 PRC04700
C                                                                       PRC04710
   25 IF(KN) 27,28,27                                                   PRC04720
   27 PHI(1)=PHI(1)*CSTG                                                PRC04730
      PHI(3)=PHI(3)*CSTA                                                PRC04740
      PHI(4)=PHI(4)*CSTI                                                PRC04750
      WRITE( 6,29) SS,(PHI(I),I=1,K)                                    PRC04760
```

```
   29 FORMAT(8H PATTERN,E15.7,8X,F6.2,12X,F6.2,12X,F6.1,14X,F6.3)      PRO04770
      PHI(1)=PHI(1)/CSTG                                                PRO04780
      PHI(3)=PHI(3)/CSTA                                                PRO04790
      PHI(4)=PHI(4)/CSTI                                                PRO04800
C                                                                       PRO04810
   28 IF(SS.GE.SPSI) GO TO 1                                            PRO04820
      DO 26 I=1,K                                                       PRO04830
      IF(ABS(PHI(I)-PSI(I)).GT.0.5*ABS(SLC(I))) GO TO 2                 PRO04840
   26 CONTINUE                                                          PRO04850
C                                                                       PRO04860
    3 IF(DELCAP.LT.DELLC) GO TO 52                                      PRO04870
      DELCAP = RHO * DELCAP                                             PRO04880
      DO 30 I=1,K                                                       PRO04890
   30 SLC(I) = RHO * SLC(I)                                             PRO04900
      GO TO 1                                                           PRO04910
C                                                                       PRO04920
   50 KONVRG = -1                                                       PRO04930
      GO TO 55                                                          PRO04940
   51 KONVRG = 0                                                        PRO04950
      GO TO 55                                                          PRO04960
   52 SPSI=S(PSI)                                                       PRO04970
      EVAL=EVAL+1                                                       PRO04980
      KONVRG = EVAL                                                     PRO04990
   55 WRITE(6,209)NUMBL,KONVRG                                          PRO05000
  209 FORMAT(/' END OF'I2,' BLADE SEARCH...'I10,' FUNCTION EVALUATIONS PPRO05010
     1ERFORMED')                                                        PRO05020
   53 DO 80 I=1,4                                                       PRO05030
   80 COORD(NUMBL,I)=PSI(I)                                             PRO05040
      COORDF(NUMBL)=SPSI                                                PRO05050
   54 RETURN                                                            PRO05060
      END                                                               PRO05070

      SUBROUTINE COMP(VJ,VKTS,ALT,FSUBT,FSUBB)                          PRO05080
C                                                                       PRO05090
C     SUBROUTINE COMP CALCULATES COMPRESSIBILITY AND BLOCKING           PRO05100
C     EFFECTS ON PROPELLER PERFORMANCE                                  PRO05110
C                                                                       PRO05120
      COMMON STATCP(28,4,8,3),STATCT(28,4,8,3),VJV(7),BETA(4,4,4,5,3),
     AFLTCPE(6,7,4,5,3),FLTETA(6,7,4,5,3),FLTCPB(4,4,4,5,3),NCP(3),CSTA,
     BWT,WAREA,NUMENG,NPROP,AR,CDPCR,CDPTO,E,FUEL,VTCKTS,NTYPE,CSTI,ATC,
     CBHPTO,RPMTC,TJETTO.BHPCL,RPMCL,TJETCL,BHPCR,RPMCR,TJETCR,BHPVM,
     DRPMVM,TJETVM,FCON,GR,ALTTO,ALTCR,ALTCL,ALTVM,AVCL,AVCR,AVM,TOCONT,
     ECLCONT,RACONT,ENCONT,CRCONT,VMCONT,DMAX,NUMBL,D,AF,CLI.CSTG,VOITO,
     FVOICL,VOIRA,VOIEN,VOICR,VOIVM,TCDIST,CTTO,RHOSL,ARC,G,NB,AFVST(8),
     GCLIV(4),NAF(3),NCLI(3),CT(9),AFVFLT(5),NJ(3),ARANGE,ETAV(27),VMLT,
     HVJVB(4),BETAV(27),RPM,BHPCR1,BHPCR2,BHPM,BHPTO1,BHPTO2,BHPTO3,NN,
     ITJETM1,TJETM2,TJETC1,TJETC2,FCON1,FCON2,BHPLIM,AENDUR,VMAX,BETAST,
```

```
JETAVM,BETAVM,VCRKTS,ETACR,BETACR,RANGE,ENDUR,RC,VCLKTS,ETACL,MM,
KBETACL,GRS,DS,AFS,CLIS,TOFAC,CRFAC,VMFAC,RAFAC,ENFAC,CLFAC,EFFM,
LDELMV(4),VJVFT(5),EMACHV(4,5),FSUBTV(4,5),VJVFB(7),FSUBBV(7),TCLT,
MEPAV(4,5),EPAALT(4,5),SHPV(5),NUMBLS,VCRLT,COORD(4,4),COORDF(4)
      DIMENSION M(3),F(3)                                               PRO05290
      IF(ALT.GT.35332.)FSUBC=.869                                       PRO05300
      IF(ALT.LE.35332.)FSUBC=1.-3.71E-06*ALT                            PRO05310
      VMACH=FSUBC*VKTS/662.                                             PRO05320
      CLI=CLI*CSTI                                                      PRO05330
      CALL NONLIN(CLIV(NCLI(1)),CLIV(NCLI(2)),CLIV(NCLI(3)),DELMV(NCLI(1PRO05340
     1)),DELMV(NCLI(2)),DELMV(NCLI(3)),CLI,DELM)                        PRO05350
      CLI=CLI/CSTI                                                      PRO05360
      EFFM=VMACH+DELM                                                   PRO05370
      DO 1 I=1,4                                                        PRO05380
      IF(VJ.LT.VJVFT(1))GO TO 2                                         PRO05390
      IF(VJ.GE.VJVFT(I).AND.VJ.LE.VJVFT(I+1))GO TO 2                    PRO05400
    1 CONTINUE                                                          PRO05410
    2 IF(I.EQ.4)GO TO 3                                                 PRO05420
      NJ(1)=I                                                           PRO05430
      NJ(2)=I+1                                                         PRO05440
      NJ(3)=I+2                                                         PRO05450
      GO TO 4                                                           PRO05460
    3 NJ(1)=I+1                                                         PRO05470
      NJ(2)=I                                                           PRO05480
      NJ(3)=I-1                                                         PRO05490
    4 DO 5 I=1,3                                                        PRO05500
      J=1                                                               PRO05510
      IF(EFFM.LT.EMACHV(1,NJ(I)))GO TO 6                                PRO05520
      DO 7 J=1,3                                                        PRO05530
      IF(EFFM.GE.EMACHV(J,NJ(I)).AND.EFFM.LE.EMACHV(J+1,NJ(I)))GO TO 6  PRO05540
    7 CONTINUE                                                          PRO05550
    6 IF(J.EQ.3)GO TO 8                                                 PRO05560
      M(1)=J                                                            PRO05570
      M(2)=J+1                                                          PRO05580
      M(3)=J+2                                                          PRO05590
      GO TO 9                                                           PRO05600
    8 M(1)=J+1                                                          PRO05610
      M(2)=J                                                            PRO05620
      M(3)=J-1                                                          PRO05630
    9 CALL NONLIN(EMACHV(M(1),NJ(I)),EMACHV(M(2),NJ(I)),EMACHV(M(3),NJ(IPRO05640
     1)),FSUBTV(M(1),NJ(I)),FSUBTV(M(2),NJ(I)),FSUBTV(M(3),NJ(I)),      PRO05650
     2EFFM,F(I))                                                        PRO05660
      IF(F(I).GT.1.)F(I)=1.                                             PRO05670
    5 CONTINUE                                                          PRO05680
      CALL NONLIN(VJVFT(NJ(1)),VJVFT(NJ(2)),VJVFT(NJ(3)),F(1),F(2),F(3),PRO05690
     1VJ,FSUBT)                                                         PRO05700
      IF(FSUBT.GT.1.)FSUBT=1.                                           PRO05710
      DO 10 I=1,6                                                       PRO05720
```

```
      IF(VJ.GE.VJVFB(I).AND.VJ.LE.VJVFB(I+1))GO TO 11                   PRO05730
   10 CONTINUE                                                          PRO05740
   11 IF(I.EQ.6)GO TO 12                                                PRO05750
      NJ(1)=I                                                           PRO05760
      NJ(2)=I+1                                                         PRO05770
      NJ(3)=I+2                                                         PRO05780
      GO TO 13                                                          PRO05790
   12 NJ(1)=I+1                                                         PRO05800
      NJ(2)=I                                                           PRO05810
      NJ(3)=I-1                                                         PRO05820
   13 CALL NONLIN(VJVFB(NJ(1)),VJVFB(NJ(2)),VJVFB(NJ(3)),FSUBBV(NJ(1)), PRO05830
     1FSUBBV(NJ(2)),FSUBBV(NJ(3)),VJ,FSUBB)                             PRO05840
      FSUBB=.94*FSUBB                                                   PRO05850
      RETURN                                                            PRO05860
      END                                                               PRO05870

                                                                        INP00010
      SUBROUTINE INPUT
      COMMON STATCP(28,4,8,3),STATCT(28,4,8,3),VJV(7),BETA(4,4,4,5,3),
     AFLTCPE(6,7,4,5,3),FLTETA(6,7,4,5,3),FLTCPB(4,4,4,5,3),NCP(3),CSTA,
     BWT,WAREA,NUMENG,NPROP,AR,CDPCR,CDPTO,E,FUEL,VTCKTS,NTYPE,CSTI,ATO,
     CBHPTO,RPMTO,TJETTC.BHPCL,RPMCL,TJETCL,BHPCR,RPMCR,TJETCR,BHPVM,
     DRPMVM,TJETVM,FCON,GR,ALTTO,ALTCR,ALTCL,ALTVM,AVCL,AVCR,AVM,TOCONT,
     ECLCONT,RACONT,ENCONT,CRCONT,VMCONT,DMAX,NUMBL,D,AF,CLI,CSTG,VOITO,
     FVOICL,VOIRA,VOIEN,VOICR,VOIVM,TODIST,CTTO,RHOSL,ARC,G,NB,AFVST(8),
     GCLIV(4),NAF(3),NCLI(3),CT(9),AFVFLT(5),NJ(3),ARANGE,ETAV(27),VMLT,
     HVJVB(4),BETAV(27),RPM,BHPCR1,BHPCR2,BHPM,BHPTO1,BHPTO2,BHPTO3,NN,
     ITJETM1,TJETM2,TJETC1,TJETC2,FCON1,FCON2,BHPLIM,AENDUR,VMAX,BETAST,
     JETAVM,BETAVM,VCRKTS,ETACR,BETACR,RANGE,ENDUR,RC,VCLKTS,ETACL,MM,
     KBETACL,GRS,DS,AFS,CLIS,TOFAC,CRFAC,VMFAC,RAFAC,ENFAC,CLFAC,EFFM,
     LDELMV(4),VJVFT(5),EMACHV(4,5),FSUBTV(4,5),VJVFB(7),FSUBBV(7),TOLT,
     MEPAV(4,5),EPAALT(4,5),SHPV(5),NUMBLS,VCRLT,COORD(4,4),COORDF(4)
                                                                        INP00180
      DIMENSION ARG(4),NHP(2),NALT(4)                                   INP00190
      NAMELIST/FTBHP/BHP,CON                                            INP00200
      TOCONT=0.                                                         INP00210
      CLCONT=0.                                                         INP00220
      VMCONT=0.                                                         INP00230
      CRCONT=0.                                                         INP00240
      RACONT=0.                                                         INP00250
      ENCONT=0.                                                         INP00260
      VOITO=0.                                                          INP00270
      VOICL=0.                                                          INP00280
      VOICR=0.                                                          INP00290
      VOIVM=0.                                                          INP00300
      VOIRA=0.                                                          INP00310
      VOIEN=0.                                                          INP00320
   16 WRITE(6,1000)                                                     INP00330
 1000 FORMAT(' THIS PROGRAM OPTIMIZES THE PROPELLER FOR A GIVEN AIRCRAFT
```

```
     1 UNDER SPECIFIED CONDITIONS',//' PLEASE TYPE IN THE REQUESTED INFOINP00340
     2RMATION, FOLLOWING THE DIRECTIONS CAREFULLY'//' INPUT ROUTINE...' INP00350
     3/10X,'STEP 1:  AIRCRAFT DATA'/,10X,'STEP 2:  ENGINE DATA'/,10X,   INP00360
     4'STEP 3:  AIRCRAFT ENVIRONMENT AND ANTICIPATED PERFORMANCE'/,10X, INP00370
     5'STEP 4:  AIRCRAFT PERFORMANCE CONSTRAINTS  (OPTIMIZATION ONLY)'/,INP00380
     610X,'STEP 5:  PROPELLER CONSTRAINTS'/,10X,'STEP 6:  OPTIMIZATION CINP00390
     7RITERIA  (OPTIMIZATION ONLY)'/,10X,'STEP 7:  END OF INPUT ROUTINE.INP00400
     8..PROPELLER EVALUATION OR OPTIMIZATION')                          INP00410
    1 WRITE(6,1001)                                                     INP00420
 1001 FORMAT(//' STEP 1: AIRCRAFT DATA'                                 INP00430
     1,/' PLEASE TYPE IN AIRCRAFT DATA AS CALLED FOR IN THE PROPER UNITSINP00440
     2....INCLUDE DECIMAL',//' DESIGN GROSS WEIGHT = ?   (LBS.)')       INP00450
      READ(5,1002)WT                                                    INP00460
 1002 FORMAT(F10.3)                                                     INP00470
      WRITE(6,1003)                                                     INP00480
 1003 FORMAT(/' WING AREA = ?    (SQ. FT.)')                            INP00490
      READ(5,1002)WAREA                                                 INP00500
      WRITE(6,1004)                                                     INP00510
 1004 FORMAT(/' NUMBER OF ENGINES = ?')                                 INP00520
      READ(5,1005)NUMENG                                                INP00530
 1005 FORMAT(1I1)                                                       INP00540
      WRITE(6,1007)                                                     INP00550
 1007 FORMAT(/' ASPECT RATIO = ?')                                      INP00560
      READ(5,1002)AR                                                    INP00570
 1096 FORMAT(/' PARASITIC DRAG COEFFICIENT = ? (TAKEOFF CONFIGURATION)')INP00580
      WRITE(6,1096)                                                     INP00590
      READ(5,1002)CDPTO                                                 INP00600
      WRITE(6,1008)                                                     INP00610
 1008 FORMAT(/' PARASITIC DRAG COEFFICIENT = ?  (CRUISE CONFIGURATION)')INP00620
      READ(5,1002)CDPCR                                                 INP00630
      WRITE(6,1009)                                                     INP00640
 1009 FORMAT(/' WING EFFICIENCY FACTOR = ?  (SMALL E)')                 INP00650
      READ(5,1002)E                                                     INP00660
      WRITE(6,1010)                                                     INP00670
 1010 FORMAT(/' FUEL CAPACITY = ?   (LBS.)')                            INP00680
      READ(5,1002)FUEL                                                  INP00690
      WRITE(6,1011)                                                     INP00700
 1011 FORMAT(/' TAKEOFF SPEED = ?    (KTS TAS)')                        INP00710
      READ(5,1002)VTOKTS                                                INP00720
      WRITE(6,1012)WT,WAREA,NUMENG,AR,CDPTO,CDPCR,E,FUEL,VTOKTS         INP00730
 1012 FORMAT(//' END OF STEP 1....AIRCRAFT DATA FOLLOW:        (CHECK VAINP00740
     1LUES CAREFULLY FOR ACCURACY)',//' WEIGHT ='F10.1,' LBS.',T40,' WININP00750
     2G AREA ='F10.1,' SQ. FT.',//' NUMBER OF ENGINES ='I2,T40,' ASPECT INP00760
     3RATIO ='F10.2,//' PARASITIC DRAG COEF. (TAKEOFF) ='F6.3,T40,' PARAINP00770
     4SITIC DRAG COEF. (CRUISE) ='F6.3,//' WING EFFICIENCY FACTOR ='    INP00780
     5F10.3,T40,' FUEL CAPACITY ='F10.1,' LBS.',//' TAKEOFF SPEED ='    INP00790
     6F10.0,' KTS TAS')                                                 INP00800
  103 WRITE(6,1040)                                                     INP00810
```

```
1040 FORMAT(//' AT THIS POINT, TYPE "0." IF YOU WISH TO CONTINUE'/      INPO0820
    1' IF YOU WISH TO RETURN OR ADVANCE TO ANOTHER STEP, TYPE NUMBER OFINPO0830
    2 STEP')                                                            INPO0840
     READ(5,1005)NSTEP                                                  INPO0850
     GO TO (1,2,3,4,5,6,7),NSTEP                                        INPO0860
   2 WRITE(6,1015)                                                      INPO0870
1015 FORMAT(//' STEP 2: ENGINE DATA'/' PLEASE TYPE IN REQUIRED ENGINE DINPO0880
    1ATA IN PROPER UNITS....ALWAYS INCLUDE A DECIMAL'/' ENTER ENGINE CHINPO0890
    2ARACTERISTICS AT WHATEVER AIR DENSITY THE AIRCRAFT PERFORMANCE IS INPO0900
    3TO BE OPTIMIZED,'/' FOR THAT PARTICULAR PORTION OF ITS MISSION'// INPO0910
    4' RECIPROCATING (1) OR TURBO-PROP (2) ENGINE ?....(TYPE NUMBER CORINPO0920
    5RESPONDING TO CORRECT CHOICE)')                                    INPO0930
     READ(5,1005)NTYPE                                                  INPO0940
     IF(NTYPE.EQ. 1)GO TO 101                                           INPO0950
     WRITE(6,1016)                                                      INPO0960
1016 FORMAT(/' ENGINE OPERATING RPM = ?')                               INPO0970
     READ(5,1002)RPM                                                    INPO0980
     RPMVM=RPM                                                          INPO0990
1045 FORMAT(/' GEAR RATIO = ?  (ENGINE RPM/PROP RPM)')                  INPO1000
     WRITE(6,1017)                                                      INPO1010
1017 FORMAT(/' SHP (PER ENGINE) AT CRUISE ALTITUDE AND CRUISE POWER SETINPO1020
    1TING,'/' FOR ANTICIPATED CRUISE SPEED (TAS) AND 50 KTS ABOVE THIS INPO1030
    2SPEED, RESPECTIVELY, = ? (HIT RETURN BETWEEN ENTRIES)')            INPO1040
     READ(5,1002)BHPCR1                                                 INPO1050
     READ(5,1002)BHPCR2                                                 INPO1060
     WRITE(6,1018)                                                      INPO1070
1018 FORMAT(/' ENTER SHP AT MAX. POWER SETTING (POWER FOR TAKEOFF AND MINPO1080
    1AX. SPEED)'/' FOR ANTICIPATED MAX. SPEED (TAS) AND 10,000 FT MSL AINPO1090
    2LTITUDE'/' SHP AT ABOVE CONDITIONS = ?')                           INPO1100
     READ(5,1002)BHPM                                                   INPO1110
     WRITE(6,1019)                                                      INPO1120
1019 FORMAT(/' ENTER SEA LEVEL SHP FOR MAX. POWER SETTING AT 0., 100, AINPO1130
    1ND 400 KTS (TAS), RESPECTIVELY'/' SHP UNDER ABOVE CONDITIONS = ? (INPO1140
    2HIT RETURN AFTER EACH SHP ENTRY)')                                 INPO1150
     READ(5,1002)BHPTO1                                                 INPO1160
     READ(5,1002)BHPTO2                                                 INPO1170
     READ(5,1002)BHPTO3                                                 INPO1180
     WRITE(6,1020)                                                      INPO1190
1020 FORMAT(/' SEA LEVEL JET THRUST (LBS PER ENGINE) FOR MAX. POWER SETINPO1200
    1TING AT 0. AND 100. KTS (TAS), RESPECTIVELY, = ?'/' (HIT RETURN AFINPO1210
    2TER EACH ENTRY)')                                                  INPO1220
     READ(5,1002)TJETM1                                                 INPO1230
     READ(5,1002)TJETM2                                                 INPO1240
     WRITE(6,1021)                                                      INPO1250
1021 FORMAT(/' SEA LEVEL JET THRUST FOR CRUISE POWER SETTING AT 0. AND INPO1260
    1100. KTS (TAS), RESPECTIVELY, = ? (HIT RETURN AFTER EACH ENTRY)') INPO1270
     READ(5,1002)TJETC1                                                 INPO1280
     READ(5,1002)TJETC2                                                 INPO1290
```

```
      WRITE(6,1022)
 1022 FORMAT(/' ENTER FUEL FLOW (LBS/HR) FOR CRUISE POWER AND ALTITUDE A
     1T ANTICIPATED CRUISE SPEED (TAS)'/' AND 50 KTS ABOVE THIS SPEED, R
     2ESPECTIVELY'/' FUEL FLOW AT ABOVE CONDITIONS = ? (HIT RETURN AFTER
     3 EACH ENTRY)')
      READ(5,1002)FCON1
      READ(5,1002)FCON2
      WRITE(6,1098)
 1098 FORMAT(/' LIMITING SHP FOR CONTINUOUS OPERATION = ?')
      READ(5,1002)BHPLIM
      WRITE(6,1023)RPM,BHPCR1,BHPCR2,BHPM ,BHPTO1,BHPTO2,BHPTO3,TJETM1,
     1TJETM2,TJETC1,TJETC2,FCON1,FCON2,BHPLIM
 1046 FORMAT(/'+GEAR RATIO ='F7.2,'  (ENGINE RPM/PROP RPM)')
 1047 FORMAT( ' ....(UNSPECIFIED)')
 1023 FORMAT(//' END OF STEP 2....ENGINE DATA (PER ENGINE) FOLLOW
     A(CHECK VALUES CAREFULLY FOR ACCURACY)'//' ENGINE OPERATING RPM ='
     BF10.1//' SHP AT CRUISE POWER AND ALTITUDE ='F10.1,' AT CRUISE SPD
     CAND ='F10.1,' AT CRUISE SPD + 50 KTS'//' SHP AT MAX. SPEED POWER A
     DND 10,000 FT ALTITUDE FOR ANTICIPATED MAX. SPEED ='F10.1//' SEA LE
     EVEL SHP AT MAX. POWER ='F10.1,' AT 0. KTS,'F10.1,' AT 100 KTS,'F10
     F.1,' AT 400 KTS'//' SEA LEVEL JET THRUST FOR MAX. POWER = 'F8.1,'
     GLBS AT 0. KTS,'F8.1,' LBS AT 100 KTS'//' SEA LEVEL JET THRUST FOR
     HCRUISE POWER ='F8.1,' LBS AT 0. KTS,'F8.1,' LBS AT 100 KTS'//' FUE
     IL FLOW FOR CRUISE POWER AND ALTITUDE ='F8.1,' LBS/HR AT CRUISE SPE
     JED,'F8.1,' LBS/HR AT CRUISE SPEED + 50 KTS'//' LIMITING SHP FOR CO
     KNTINUOUS OPERATION ='F10.1)
      WRITE(6,1040)
      READ(5,1005)NSTEP
      GO TO (1,2,3,4,5,6,7),NSTEP
      GO TO 3
  101 WRITE(6,1029)
 1029 FORMAT(/' ENGINE PERFORMANCE AT TAKEOFF (PER ENGINE) :'//' BRAKE H
     1ORSEPOWER = ?')
      READ(5,1002)BHPTO
      WRITE(6,1024)
 1024 FORMAT(/' ENGINE RPM AT TAKEOFF (ZERO FORWARD SPEED) = ?')
      READ(5,1002)RPMTO
      WRITE(6,1030)
 1030 FORMAT(/' ENGINE PERFORMANCE AT MAXIMUM CLIMB RATE (PER ENGINE) :'
     1//' BHP = ?')
      READ(5,1002)BHPCL
      WRITE(6,1025)
 1025 FORMAT(/' ENGINE RPM AT MAX. CLIMB = ?')
      READ(5,1002)RPMCL
      WRITE(6,1031)
 1031 FORMAT(/' ENGINE PERFORMANCE AT CRUISE CONDITIONS (PER ENGINE) :'
     1//' BHP = ?')
      READ(5,1002)BHPCR
```

```
      WRITE(6,1027)                                                     INP01780
 1027 FORMAT(/' ENGINE RPM AT CRUISE = ?')                              INP01790
      READ(5,1002)RPMCR                                                 INP01800
      WRITE(6,1032)                                                     INP01810
 1032 FORMAT(/' ENGINE PERFORMANCE AT MAXIMUM SPEED (PER ENGINE) :'     INP01820
     1//' BHP = ?')                                                     INP01830
      READ(5,1002)BHPVM                                                 INP01840
      WRITE(6,1026)                                                     INP01850
 1026 FORMAT(/' ENGINE RPM AT MAX. SPEED = ?')                          INP01860
      READ(5,1002)RPMVM                                                 INP01870
      WRITE(6,1034)                                                     INP01880
      READ(5,1002)FCON                                                  INP01890
      WRITE(6,1033)BHPTO,RPMTO,BHPCL,RPMCL,BHPCR,RPMCR,BHPVM,RPMVM,FCON INP01900
      WRITE(6,1040)                                                     INP01910
      READ(5,1005)NSTEP                                                 INP01920
      GO TO (1,2,3,4,5,6,7),NSTEP                                       INP01930
 1033 FORMAT (//' END OF STEP 2....ENGINE DATA (PER ENGINE) FOLLOW      INP01940
     1(CHECK VALUES CAREFULLY FOR ACCURACY)'//' BHP (TAKEOFF) ='F10.1, INP01950
     2T50,' RPM (TAKEOFF) ='F10.1,//' BHP (MAX. CLIMB) ='F10.1,T50,    INP01960
     3' RPM (MAX. CLIMB) ='F10.1,//' BHP (CRUISE) ='F10.1,T50,         INP01970
     4' RPM (CRUISE) ='F10.1,//' BHP (MAX. SPEED) ='F10.1,T50,         INP01980
     5' RPM (MAX. SPEED) ='F10.1//' FUEL CONSUMPTION (CRUISE) ='F10.1,  INP01990
     6' LBS/HR')                                                        INP02000
 1034 FORMAT(/' FUEL CONSUMPTION (PER ENGINE) AT CRUISE POWER, SPEED, ANINP02010
     1D ALTITUDE = ?  (LBS/HR)')                                        INP02020
    3 WRITE(6,1048)                                                     INP02030
 1048 FORMAT(//' STEP 3:  AIRCRAFT ENVIRONMENT AND ANTICIPATED PERFORMANINP02040
     1CE'/' PLEASE TYPE IN REQUIRED DATA IN PROPER UNITS....ALWAYS INCLUINP02050
     2DE A DECIMAL'//' PLEASE TYPE IN ALTITUDES (FT. MSL) AT WHICH AIRCRINP02060
     3AFT PERFORMANCE IS TO BE EVALUATED'/' AT TAKEOFF, CLIMB,CRUISE, ANINP02070
     4D MAXIMUM SPEED....(SHOULD CORRESPOND WITH ENGINE DATA)'//' ALTITUINP02080
     5DE AT TAKEOFF = ?  (FT. MSL)')                                    INP02090
      READ(5,1002)ALTTO                                                 INP02100
      WRITE(6,1049)                                                     INP02110
 1049 FORMAT(/' ALTITUDE AT CLIMB = ?')                                 INP02120
      READ(5,1002)ALTCL                                                 INP02130
      WRITE(6,1050)                                                     INP02140
 1050 FORMAT(/' CRUISE ALTITUDE = ?')                                   INP02150
      READ(5,1002)ALTCR                                                 INP02160
      WRITE(6,1051)                                                     INP02170
 1051 FORMAT(/' ALTITUDE AT MAX. SPEED = ?')                            INP02180
      READ(5,1002)ALTVM                                                 INP02190
      WRITE(6,1052)                                                     INP02200
 1052 FORMAT(/' ANTICIPATED AIRCRAFT SPEEDS (TAS):'//' CLIMB SPEED = ?  INP02210
     1(KTS)')                                                           INP02220
      READ(5,1002)AVCL                                                  INP02230
      WRITE(6,1053)                                                     INP02240
 1053 FORMAT(/' CRUISE SPEED = ?  (KTS)')                               INP02250
```

```
                                                                        INP02260
      READ(5,1002)AVCR                                                  INP02270
      WRITE(6,1054)                                                     INP02280
 1054 FORMAT(/' MAX. SPEED = ?  (KTS)')                                 INP02290
      READ(5,1002)AVM                                                   INP02300
      WRITE(6,1055)ALTTO,ALTCL,ALTCR,ALTVM,AVCL,AVCR,AVM                INP02310
 1055 FORMAT(//' END OF STEP 3....ENVIRONMENT AND ANTICIPATED SPEED DATAINP02320
     1 FOLLOW  (CHECK VALUES CAREFULLY FOR ACCURACY)'//' ALTITUDE AT TAKINP02330
     2EOFF ='F10.1,' FT MSL',T50,' ALTITUDE AT CLIMB ='F10.1,' FT MSL'//INP02340
     3' CRUISE ALTITUDE ='F10.1,' FT MSL',T50,' ALTITUDE AT MAX. SPEED =INP02350
     4'F10.1,' FT MSL',//' CLIMB SPEED ='F7.1,' KTS (TAS)',T40,'CRUISE SINP02360
     5PEED ='F7.1,' KTS (TAS)',T80,' MAX. SPEED ='F7.1,' KTS (TAS)')    INP02370
      WRITE(6,1040)                                                     INP02380
      READ(5,1005)NSTEP                                                 INP02390
      GO TO (1,2,3,4,5,6,7),NSTEP                                       INP02400
    4 WRITE(6,1056)                                                     INP02410
 1056 FORMAT(//' STEP 4:  PERFORMANCE CONSTRAINTS'/' PLEASE TYPE IN REQUINP02420
     1IRED DATA IN PROPER UNITS....ALWAYS INCLUDE A DECIMAL'//' TYPE "0.INP02430
     2" AS THE VALUE OF ANY CONSTRAINT WHICH YOU DO NOT WISH TO SPECIFY'INP02440
     3//' MAXIMUM ALLOWABLE TAKEOFF DISTANCE = ?  (FT)')                INP02450
      READ(5,1002)TOCONT                                                INP02460
      WRITE(6,1057)                                                     INP02470
 1057 FORMAT(/' LOWEST ALLOWABLE MAXIMUM RATE OF CLIMB = ?  (FT/MIN)')  INP02480
      READ(5,1002)CLCONT                                                INP02490
      WRITE(6,1058)                                                     INP02500
 1058 FORMAT(/' MINIMUM ALLOWABLE RANGE AT CRUISE CONDITIONS = ?(N MI)')INP02510
      READ(5,1002)RACONT                                                INP02520
      WRITE(6,1059)                                                     INP02530
 1059 FORMAT(/' MINIMUM ALLOWABLE ENDURANCE AT CRUISE CONDITIONS = ?  (HINP02540
     1RS)')                                                             INP02550
      READ(5,1002)ENCONT                                                INP02560
      WRITE(6,1060)                                                     INP02570
 1060 FORMAT(/' MINIMUM ALLOWABLE CRUISE SPEED = ?  (KTS TAS)')         INP02580
      READ(5,1002)CRCONT                                                INP02590
      WRITE(6,1061)                                                     INP02600
 1061 FORMAT(/' LOWEST ALLOWABLE MAXIMUM SPEED = ?  (KTS TAS)')         INP02610
      READ(5,1002)VMCONT                                                INP02620
      WRITE(6,1062)                                                     INP02630
 1062 FORMAT(//' END OF STEP 4....PERFORMANCE CONSTRAINTS FOLLOW   (CHECINP02640
     1K VALUES CAREFULLY FOR ACCURACY)')                                INP02650
      WRITE(6,1063)TOCONT                                               INP02660
      IF(TOCONT.EQ.0.)WRITE(6,1047)                                     INP02670
      WRITE(6,1064)CLCONT                                               INP02680
      IF(CLCONT.EQ.0.)WRITE(6,1047)                                     INP02690
      WRITE(6,1065)RACONT                                               INP02700
      IF(RACONT.EQ.0.)WRITE(6,1047)                                     INP02710
      WRITE(6,1066)ENCONT                                               INP02720
      IF(ENCONT.EQ.0.)WRITE(6,1047)                                     INP02730
      WRITE(6,1067)CRCONT
```

```
      IF(CRCONT.EQ.0.)WRITE(6,1047)                                     INP02740
      WRITE(6,1068)VMCONT                                               INP02750
      IF(VMCONT.EQ.0.)WRITE(6,1047)                                     INP02760
 1063 FORMAT(/'+MAX. ALLOWABLE TAKEOFF DISTANCE ='F8.1,' FT/MIN')       INP02770
 1064 FORMAT(/'+LOWEST ALLOWABLE MAX. RATE OF CLIMB ='F7.1,' FT/MIN')   INP02780
 1065 FORMAT(/'+MINIMUM ALLOWABLE RANGE AT CRUISE CONDITIONS ='F8.1,' N INP02790
     1MI')                                                              INP02800
 1066 FORMAT(/'+MINIMUM ALLOWABLE ENDURANCE AT CRUISE CONDITIONS ='     INP02810
     1F5.1,' HRS')                                                      INP02820
 1067 FORMAT(/'+MINIMUM ALLOWABLE CRUISE SPEED ='F6.1,' KTS')           INP02830
 1068 FORMAT(/'+LOWEST ALLOWABLE MAX. SPEED ='F6.1,' KTS')              INP02840
      WRITE(6,1040)                                                     INP02850
      READ(5,1005)NSTEP                                                 INP02860
      GO TO (1,2,3,4,5,6,7),NSTEP                                       INP02870
    5 WRITE(6,1069)                                                     INP02880
 1069 FORMAT(//' STEP 5:  PROPELLER CONSTRAINTS'/' PLEASE TYPE IN REQUIRINP02890
     1ED DATA IN PROPER UNITS....ALWAYS INCLUDE A DECIMAL'//' MAX. BLADEINP02900
     2 DIAMETER FOR STRUCTURAL OR GROUND CLEARANCE = ?  (FT)')          INP02910
      READ(5,1002)DMAX                                                  INP02920
      WRITE(6,1084)DMAX                                                 INP02930
      WRITE(6,1070)                                                     INP02940
 1084 FORMAT(/' MAX. BLADE DIAMETER ='F6.2,' FT')                       INP02950
 1070 FORMAT(/' THE FOLLOWING PROP CONSTRAINTS ARE OPTIONAL'/           INP02960
     1' IF NOT SPECIFIED, THE OPTIMUM VALUE OF EACH QUANTITY WILL BE DETINP02970
     2ERMINED'/' TYPE "0." AS THE VALUE OF ANY CONSTRAINT WHICH YOU DO NINP02980
     3OT WISH TO SPECIFY')                                              INP02990
   14 WRITE(6,1071)                                                     INP03000
 1071 FORMAT(/' NUMBER OF BLADES = ?')                                  INP03010
      READ(5,1005)NUMBLS                                                INP03020
      IF(NUMBLS.NE.0.AND.(NUMBLS.LT.2.OR.NUMBLS.GT.4))GO TO 10          INP03030
      WRITE(6,1045)                                                     INP03040
      READ(5,1002)GRS                                                   INP03050
      WRITE(6,1072)                                                     INP03060
 1072 FORMAT(/' BLADE DIAMETER = ?  (FT)')                              INP03070
      READ(5,1002)DS                                                    INP03080
      WRITE(6,1074)                                                     INP03090
 1074 FORMAT(/' ACTIVITY FACTOR = ?')                                   INP03100
      READ(5,1002)AFS                                                   INP03110
      IF(AFS.NE.0..AND.((NUMBLS.EQ.2.AND.AFS.GT.120.).OR.(AFS.LT.80..OR.INP03120
     1AFS.GT.220.)))GO TO 11                                            INP03130
      WRITE(6,1075)                                                     INP03140
 1075 FORMAT(/' INTEGRATED LIFT COEFFICIENT = ?')                       INP03150
      READ(5,1002)CLIS                                                  INP03160
      IF(CLIS.NE.0..AND.((NUMBLS.EQ.2.AND.CLIS.LT..3).OR.(CLIS.LT..15   INP03170
     1.OR.CLIS.GT..7)))GO TO 12                                         INP03180
      WRITE(6,1077)                                                     INP03190
 1077 FORMAT(//' END OF STEP 5....OPTIONAL BLADE CONSTRAINTS FOLLOW     INP03200
     1(CHECK VALUES CAREFULLY FOR ACCURACY)')                           INP03210
```

```
      WRITE(6,1046)GRS                                                  INP03220
      IF(GRS.EQ.0.)WRITE(6,1047)                                        INP03230
      WRITE(6,1078)NUMBLS                                               INP03240
      IF(NUMBLS.EQ.0)WRITE(6,1047)                                      INP03250
      WRITE(6,1079)DS                                                   INP03260
      IF(DS.EQ.0.)WRITE(6,1047)                                         INP03270
      WRITE(6,1081)AFS                                                  INP03280
      IF(AFS.EQ.0.)WRITE(6,1047)                                        INP03290
      WRITE(6,1082)CLIS                                                 INP03300
      IF(CLIS.EQ.0.)WRITE(6,1047)                                       INP03310
1078  FORMAT(/'+NUMBER OF BLADES ='I2)                                  INP03320
1079  FORMAT(/'+BLADE DIAMETER ='F6.2,' FT')                            INP03330
1081  FORMAT(/'+ACTIVITY FACTOR ='F7.1)                                 INP03340
1082  FORMAT(/'+INTEGRATED LIFT COEFFICIENT ='F6.3)                     INP03350
      GO TO 17                                                          INP03360
   10 WRITE(6,1094)                                                     INP03370
      GO TO 14                                                          INP03380
   11 WRITE(6,1093)                                                     INP03390
      GO TO 14                                                          INP03400
   12 WRITE(6,1095)                                                     INP03410
      GO TO 14                                                          INP03420
1094  FORMAT(/' DATA NOT AVAILABLE FOR SPECIFIED NUMBER OF BLADES')     INP03430
1093  FORMAT(/' DATA NOT AVAILABLE FOR SPECIFIED ACTIVITY FACTOR')      INP03440
1095  FORMAT(/' DATA NOT AVAILABLE FOR SPECIFIED INTEGRATED LIFT COEFFICINP03450
     1IENT')                                                            INP03460
   17 AF=AFS                                                            INP03470
      CLI=CLIS                                                          INP03480
      GR=GRS                                                            INP03490
      NUMBL=NUMBLS                                                      INP03500
      D=DS                                                              INP03510
   13 WRITE(6,1040)                                                     INP03520
      READ(5,1005)NSTEP                                                 INP03530
      GO TO(1,2,3,4,5,6,7),NSTEP                                        INP03540
    6 WRITE(6,1085)                                                     INP03550
1085  FORMAT(//' STEP 6:  OPTIMIZATION CRITERIA'/' PLEASE RATE EACH OF TINP03560
     1HE FOLLOWING PERFORMANCE TRAITS, ON A SCALE FROM 0. TO 10.,'/     INP03570
     2' ACCORDING TO RELATIVE IMPORTANCE TO AIRCRAFT MISSION'//         INP03580
     3' VALUES ASSIGNED SHOULD BE RELATIVE ONLY TO THE OTHER VALUES'/   INP03590
     4' ABSOLUTE MAGNITUDE IS ARBITRARY'//                              INP03600
     5' "VALUE OF IMPORTANCE" OF SHORT TAKEOFF DISTANCE = ?')           INP03610
      READ(5,1002)VOITO                                                 INP03620
      WRITE(6,1086)                                                     INP03630
1086  FORMAT(/' V.O.I. OF CLIMB RATE = ?')                              INP03640
      READ(5,1002)VOICL                                                 INP03650
      WRITE(6,1087)                                                     INP03660
1087  FORMAT(/' V.O.I. OF RANGE = ?')                                   INP03670
      READ(5,1002)VOIRA                                                 INP03680
      WRITE(6,1088)                                                     INP03690
```

```
 1088 FORMAT(/' V.O.I. OF ENDURANCE = ?')                                  INP03700
      READ(5,1002)VOIEN                                                    INP03710
      WRITE(6,1089)                                                        INP03720
 1089 FORMAT(/' V.O.I. OF CRUISE SPEED = ?')                               INP03730
      READ(5,1002)VOICR                                                    INP03740
      WRITE(6,1090)                                                        INP03750
 1090 FORMAT(/' V.O.I. OF MAX. SPEED = ?')                                 INP03760
      READ(5,1002)VOIVM                                                    INP03770
      WRITE(6,1091)VOITO,VOICL,VOIRA,VOIEN,VOICR,VOIVM                     INP03780
 1091 FORMAT(//' END OF STEP 6....OPTIMIZATION CRITERIA FOLLOW             INP03790
     1(CHECK VALUES CAREFULLY FOR ACCURACY)'//' V.O.I. OF SHORT TAKEOFF     INP03800
     2DISTANCE ='F6.1,/' V.O.I. OF CLIMB RATE ='F6.1,/' V.O.I. OF RANGE    INP03810
     3='F6.1,/' V.O.I. OF ENDURANCE ='F6.1,/' V.O.I. OF CRUISE SPEED =     INP03820
     4F6.1,/' V.O.I. OF MAX. SPEED ='F6.1)                                 INP03830
      WRITE(6,1040)                                                        INP03840
      READ(5,1005)NSTEP                                                    INP03850
      GO TO (1,2,3,4,5,6,7),NSTEP                                          INP03860
    7 WRITE(6,1092)                                                        INP03870
 1092 FORMAT(//' END OF INPUT ROUTINE....YOU SHOULD HAVE TIME FOR A QUICINP03880
     1K CUP OF COFFEE WHILE AWAITING RESULTS')                             INP03890
      NN=0                                                                 INP03900
      IF((GRS*DS*AFS*CLIS*FLCAT(NUMBLS)).EQ.0.)GO TO 70                    INP03910
      MM=1                                                                 INP03920
      ARG(1)=GR/CSTG                                                       INP03930
      ARG(2)=D                                                             INP03940
      ARG(3)=AF/CSTA                                                       INP03950
      ARG(4)=CLI/CSTI                                                      INP03960
      DUM=FUNCT(ARG)                                                       INP03970
      GO TO 9                                                              INP03980
   70 IF(NUMBLS.NE.0)GO TO 8                                               INP03990
      MM=0                                                                 INP04000
      NUMBL=4                                                              INP04010
    8 DELCAP=.816                                                          INP04020
      EXTERNAL FUNCT                                                       INP04030
      IF(NTYPE.EQ.2)BHPVM=BHPLIM                                           INP04040
      IF(NTYPE.EQ.2)RPMVM=RPM                                              INP04050
      DO 50 I=1,4                                                          INP04060
      IF(BHPVM.LT.SHPV(1))GO TO 51                                         INP04070
      IF(BHPVM.GE.SHPV(I).AND.BHPVM.LE.SHPV(I+1))GO TO 51                  INP04080
   50 CONTINUE                                                             INP04090
   51 NHP(1)=I                                                             INP04100
      NHP(2)=I+1                                                           INP04110
   53 DO 60 J=1,2                                                          INP04120
      DO 54 I=1,3                                                          INP04130
      IF(ALTVM.LT.EPAALT(1,NHP(J)))GO TO 55                                INP04140
      IF(ALTVM.GE.EPAALT(I,NHP(J)).AND.ALTVM.LE.EPAALT(I+1,NHP(J)))        INP04150
     1GO TO 55                                                             INP04160
   54 CONTINUE                                                             INP04170
```

```
   55 J1=J+(J-1)                                                        INP04180
      J2=2*J                                                            INP04190
      NALT(J1)=I                                                        INP04200
      NALT(J2)=I+1                                                      INP04210
   60 CONTINUE                                                          INP04220
      CALL LIN(EPAALT(NALT(1),NHP(1)),EPAALT(NALT(2),NHP(1)),EPAV(NALT(1INP04230
     1),NHP(1)),EPAV(NALT(2),NHP(1)),ALTVM,EPA1)                        INP04240
      CALL LIN(EPAALT(NALT(3),NHP(2)),EPAALT(NALT(4),NHP(2)),EPAV(NALT(3INP04250
     1),NHP(2)),EPAV(NALT(4),NHP(2)),ALTVM,EPA2)                        INP04260
      CALL LIN(EPA1,EPA2,SHPV(NHP(1)),SHPV(NHP(2)),BHPVM,EPA)           INP04270
      IF(EPA.LE.0.)EPA=1000.                                            INP04280
   61 IF(CLIS.EQ.0.)CLI=.4                                              INP04290
      IF(CLIS.NE.0.)CLI=CLIS                                            INP04300
   62 IF(AFS.EQ.0.)AF=80.                                               INP04310
      IF(AFS.NE.0.)AF=AFS                                               INP04320
      IF(DS.NE.0.)GO TO 63                                              INP04330
      D=SQRT(EPA/(FLOAT(NUMBL)*AF))                                     INP04340
      IF(D.LE.DMAX)GO TO 200                                            INP04350
      D=DMAX                                                            INP04360
   63 IF(AFS.NE.0.)GO TO 200                                            INP04370
      AF=EPA/(FLOAT(NUMBL)*(D**2))                                      INP04380
      IF((NUMBL.NE.2.AND.AF.GE.80..AND.AF.LE.220.).OR.(NUMBL.EQ.2.AND.AFINP04390
     1.GE.80..AND.AF.LE.120.))GO TO 200                                 INP04400
      IF(AF.LT.80.)GO TO 66                                             INP04410
      IF(NUMBL.EQ.2)AF=120.                                             INP04420
      IF(NUMBL.NE.2)AF=220.                                             INP04430
      GO TO 200                                                         INP04440
   66 AF=80.                                                            INP04450
  200 IF(GRS.EQ.0.)GR=(D*RPMVM)/(50.7*AVM)                              INP04460
      IF(GRS.NE.0.)GR=GRS                                               INP04470
      ARG(1)=GR/CSTG                                                    INP04480
      ARG(2)=D                                                          INP04490
      ARG(3)=AF/CSTA                                                    INP04500
      ARG(4)=CLI/CSTI                                                   INP04510
   15 CALL DIRECT(ARG,4,FMIN,DELCAP,.25,  .05,FUNCT,KONVRG,500,-1)      INP04520
      IF(NUMBLS.NE.0)GO TO 79                                           INP04530
      IF(NUMBL.NE.4)GO TO 71                                            INP04540
      NUMBL=3                                                           INP04550
      GO TO 8                                                           INP04560
   71 IF(NUMBL.EQ.2)GO TO 72                                            INP04570
      IF(COORDF(3).LE.COORDF(4))GO TO 73                                INP04580
      DO 74 I=1,4                                                       INP04590
   74 ARG(I)=COORD(4,I)                                                 INP04600
      NUMBL=4                                                           INP04610
      GO TO 75                                                          INP04620
   73 NUMBL=2                                                           INP04630
      GO TO 8                                                           INP04640
   72 IF(COORDF(2).LE.COORDF(3))GO TO 76                                INP04650
```

```
      DO 77 I=1,4                                                       INP04660
   77 ARG(I)=COORD(3,I)                                                 INP04670
      NUMBL=3                                                           INP04680
      GO TO 75                                                          INP04690
   76 DO 78 I=1,4                                                       INP04700
   78 ARG(I)=COORD(2,I)                                                 INP04710
      NUMBL=2                                                           INP04720
      GO TO 75                                                          INP04730
   79 DO 150 I=1,4                                                      INP04740
  150 ARG(I)=COORD(NUMBL,I)                                             INP04750
   75 NN=1                                                              INP04760
      MM=1                                                              INP04770
      DUM=FUNCT(ARG)                                                    INP04780
    9 WRITE(6,2000)                                                     INP04790
 2000 FORMAT(//' AT THIS POINT, IF YOU WISH TO TERMINATE PROGRAM OPERATIINP04800
     1ON, TYPE "C."'/' IF YOU WISH TO RETURN TO AN INPUT STEP, TYPE NUMBINP04810
     2ER OF STEP')                                                      INP04820
      READ(5,1005)NSTEP                                                 INP04830
      GO TO (1,2,3,4,5,6,7),NSTEP                                       INP04840
      RETURN                                                            INP04850
      END                                                               INP04860

      FUNCTION FUNCT(ARG)                                               EVA00010
      COMMON STATCP(28,4,8,3),STATCT(28,4,8,3),VJV(7),BETA(4,4,4,5,3),
     AFLTCPE(6,7,4,5,3),FLTETA(6,7,4,5,3),FLTCPB(4,4,4,5,3),NCP(3),CSTA,
     BWT,WAREA,NUMENG,NPROP,AR,CDPCR,CDPTO,E,FUEL,VTCKTS,NTYPE,CSTI,ATO,
     CBHPTO,RPMTO,TJETTO,BHPCL,RPMCL,TJETCL,BHPCR,RPMCR,TJETCR,BHPVM,
     DRPMVM,TJETVM,FCON,GR,ALTTO,ALTCR,ALTCL,ALTVM,AVCL,AVCR,AVM,TOCONT,
     ECLCONT,RACONT,ENCONT,CRCONT,VMCONT,DMAX,NUMBL,D,AF,CLI,CSTG,VOITO,
     FVOICL,VOIRA,VOIEN,VOICR,VOIVM,TODIST,CTTO,RHOSL,ARC,G,NB,AFVST(8),
     GCLIV(4),NAF(3),NCLI(3),CT(9),AFVFLT(5),NJ(3),ARANGE,ETAV(27),VMLT,
     HVJVB(4),BETAV(27),RPM,BHPCR1,BHPCR2,BHPM,BHPTO1,BHPTO2,BHPTO3,NN,
     ITJETM1,TJETM2,TJETC1,TJETC2,FCON1,FCON2,BHPLIM,AENDUR,VMAX,BETAST,
     JETAVM,BETAVM,VCRKTS,ETACR,BETACR,RANGE,ENDUR,RC,VCLKTS,ETACL,MM,
     KBETACL,GRS,DS,AFS,CLIS,TOFAC,CRFAC,VMFAC,RAFAC,ENFAC,CLFAC,EFFM,
     LDELMV(4),VJVFT(5),EMACHV(4,5),FSUBTV(4,5),VJVFB(7),FSUBBV(7),TOLT,
     MEPAV(4,5),EPAALT(4,5),SHPV(5),NUMBLS,VCRLT,COORD(4,4),COORDF(4)   EVA00180
      DIMENSION ARG(4)                                                  EVA00190
      GR=ARG(1)                                                         EVA00200
      D=ARG(2)                                                          EVA00210
      AF=ARG(3)                                                         EVA00220
      CLI=ARG(4)                                                        EVA00230
      IF(NN.EQ.0)GO TO 1                                                EVA00240
      GR=GR*CSTG                                                        EVA00250
      AF=AF*CSTA                                                        EVA00260
      CLI=CLI*CSTI                                                      EVA00270
      WRITE(6,56)NUMBL,GR,D,AF,CLI,COORDF(NUMBL)
```

```
   56 FORMAT(/' OPTIMUM PROPELLER :'//' NUMBER OF BLADES ='I2/' GEAR RATEVA00280
     1IO ='F6.2/' DIAMETER ='F6.2,' FT'/' ACTIVITY FACTOR ='F6.1/' INTEGEVA00290
     2RATED LIFT COEFFICIENT ='F6.3/' FUNCTION VALUE ='E15.7)          EVA00300
      GR=GR/CSTG                                                        EVA00310
      AF=AF/CSTA                                                        EVA00320
      CLI=CLI/CSTI                                                      EVA00330
    1 TOFAC=0.                                                          EVA00340
      CLFAC=0.                                                          EVA00350
      CRFAC=0.                                                          EVA00360
      VMFAC=0.                                                          EVA00370
      RAFAC=0.                                                          EVA00380
      ENFAC=0.                                                          EVA00390
      TOLT=0.                                                           EVA00400
      VMLT=0.                                                           EVA00410
      VCRLT=0.                                                          EVA00420
      ATO=18.*VTOKTS                                                    EVA00430
      IF(NTYPE.EQ.2)FCON=FCON1                                          EVA00440
      AENDUR=FUEL/(FLOAT(NUMENG)*FCON)-.75                              EVA00450
      ARANGE=AVCR*AENDUR                                                EVA00460
      AREQHP=.02*RHOSL*SIGMA(ALTCL)*WAREA*((1.689*AVCL)**3)/550.        EVA00470
      IF(NTYPE.EQ.2)BHPCL=BHPTO2                                        EVA00480
      AAVHP=FLOAT(NUMENG)*BHPCL*.8                                      EVA00490
      ARC=(AAVHP-AREQHP)*33000./WT                                      EVA00500
      IF(NTYPE.EQ.1)TJET=0.                                             EVA00510
      IF(NTYPE.EQ.1)GO TO 93                                            EVA00520
      CALL NONLIN(0.,100.,400.,BHPTO1,BHPTO2,BHPTO3,AVM,BHP2)           EVA00530
      CALL LIN(0.,10000.,BHP2,BHPM,ALTTO,BHP3)                          EVA00540
      ALTCOR=BHP3-BHP2                                                  EVA00550
      BHPTO=BHPTO1+ALTCOR                                               EVA00560
      TJET=TJETM1                                                       EVA00570
      IF(BHPTO.GT.BHPLIM)TJET=(1.-(BHPTO-BHPLIM)/BHPTO)*TJET            EVA00580
      IF(BHPTO.GT.BHPLIM)BHPTO=BHPLIM                                   EVA00590
      RPMTO=RPM                                                         EVA00600
   93 CPTO=BHPTO/(2000.*SIGMA(ALTTO)*((RPMTO/(1000.*GR*CSTG))**3)*((D/1CEVA00610
     1.)**5))                                                           EVA00620
      CPTO1=CPTO                                                        EVA00630
      FCOEF=.025                                                        EVA00640
      CLOPT=.5*3.141593*FCOEF*E*AR                                      EVA00650
      CD=CDPTO+((CLOPT**2 )/(3.141593*AR*E))                            EVA00660
      NB=NUMBL-1                                                        EVA00670
      IF(NUMBL.GE.3)GO TO 97                                            EVA00680
      DO 98 I=1,3                                                       EVA00690
   98 CLIV(I)=.3+.2*(FLOAT(I)-1.)                                       EVA00700
      J2=2                                                              EVA00710
      GO TO 94                                                          EVA00720
   97 CLIV(1)=.15                                                       EVA00730
      DO 99 I=2,4                                                       EVA00740
   99 CLIV(I)=.3+.2*(FLOAT(I)-1.)                                       EVA00750
```

```
      J2=3                                                              EVA00760
      IF(NUMBL-3)94,2,13                                                EVA00770
   13 DO 5 I=1,8                                                        EVA00780
    5 AFVST(I)=80.+20.*(FLOAT(I)-1.)                                    EVA00790
      J=7                                                               EVA00800
      AFVFLT(1)=80.                                                     EVA00810
      DO 42 I=2,5                                                       EVA00820
   42 AFVFLT(I)=100.+(40.*(FLOAT(I)-2.))                                EVA00830
      J1=4                                                              EVA00840
      GO TO 6                                                           EVA00850
    2 DO 3 I=1,3                                                        EVA00860
    3 AFVST(I)=80.+20.*(FLOAT(I)-1.)                                    EVA00870
      DO 4 I=4,7                                                        EVA00880
    4 AFVST(I)=80.+20.*FLOAT(I)                                         EVA00890
      J=6                                                               EVA00900
      AFVFLT(1)=80.                                                     EVA00910
      AFVFLT(2)=100.                                                    EVA00920
      AFVFLT(3)=180.                                                    EVA00930
      AFVFLT(4)=220.                                                    EVA00940
      J1=3                                                              EVA00950
      GO TO 6                                                           EVA00960
   94 DO 95 I=1,3                                                       EVA00970
   95 AFVST(I)=80.+20.*(FLOAT(I)-1.)                                    EVA00980
      J=2                                                               EVA00990
      DO 96 I=1,3                                                       EVA01000
   96 AFVFLT(I)=AFVST(I)                                                EVA01010
      J1=2                                                              EVA01020
    6 DO 8 I=1,J                                                        EVA01030
      IF((AF*CSTA).LT.AFVST(1))GO TO 9                                  EVA01040
      IF((AF*CSTA).GE.AFVST(I).AND.(AF*CSTA).LE.AFVST(I+1))GO TO 9      EVA01050
    8 CONTINUE                                                          EVA01060
    9 IF(I.EQ.J)GO TO 10                                                EVA01070
      NAF(1)=I                                                          EVA01080
      NAF(2)=I+1                                                        EVA01090
      NAF(3)=I+2                                                        EVA01100
      GO TO 11                                                          EVA01110
   10 NAF(1)=I+1                                                        EVA01120
      NAF(2)=I                                                          EVA01130
      NAF(3)=I-1                                                        EVA01140
   11 DO 14 I=1,J2                                                      EVA01150
      IF((CLI*CSTI).LT.CLIV(1))GO TO 15                                 EVA01160
      IF((CLI*CSTI).GE.CLIV(I).AND.(CLI*CSTI).LE.CLIV(I+1))GO TO 15     EVA01170
   14 CONTINUE                                                          EVA01180
   15 IF(I.EQ.J2)GO TO 18                                               EVA01190
      NCLI(1)=I                                                         EVA01200
      NCLI(2)=I+1                                                       EVA01210
      NCLI(3)=I+2                                                       EVA01220
      GO TO 19                                                          EVA01230
```

```
18    NCLI(1)=I+1                                                          EVA01240
      NCLI(2)=I                                                            EVA01250
      NCLI(3)=I-1                                                          EVA01260
19    IF(MM.EQ.0.AND.VOITO.EQ.0..AND.TCCONT.EQ.0.)GO TO 121                EVA01270
      K3=1                                                                 EVA01280
24    DO 20 K1=1,3                                                         EVA01290
      DO 21 K2=1,3                                                         EVA01300
      DO 22 I=1,27                                                         EVA01310
      IF(CPTO.LT.STATCP(1,NCLI(K2),NAF(K1),NB))GO TO 32                    EVA01320
      IF(CPTO.GE.STATCP(I,NCLI(K2),NAF(K1),NB).AND.CPTO.LE.STATCP(I+1,     EVA01330
     1NCLI(K2),NAF(K1),NB))GO TO 23                                        EVA01340
      IF(STATCP(I+1,NCLI(K2),NAF(K1),NB).EQ.0.)GO TO 87                    EVA01350
      GO TO 22                                                             EVA01360
23    CALL LIN(STATCP(I,NCLI(K2),NAF(K1),NB),STATCP(I+1,NCLI(K2),NAF(K1)   EVA01370
     1,NB),STATCT(I,NCLI(K2),NAF(K1),NB),STATCT(I+1,NCLI(K2),NAF(K1),NB)   EVA01380
     2,CPTO,CT(K3))                                                        EVA01390
      GO TO 88                                                             EVA01400
87    CALL NONLIN(STATCP(I-2,NCLI(K2),NAF(K1),NB),STATCP(I-1,NCLI(K2),     EVA01410
     1NAF(K1),NB),STATCP(I,NCLI(K2),NAF(K1),NB),STATCT(I-2,NCLI(K2),       EVA01420
     2NAF(K1),NB),STATCT(I-1,NCLI(K2),NAF(K1),NB),STATCT(I,NCLI(K2),       EVA01430
     3NAF(K1),NB),CPTO,CT(K3))                                             EVA01440
88    K3=K3+1                                                              EVA01450
      GO TO 21                                                             EVA01460
32    CT(K3)=0.                                                            EVA01470
      K3=K3+1                                                              EVA01480
      GO TO 21                                                             EVA01490
22    CONTINUE                                                             EVA01500
21    CONTINUE                                                             EVA01510
20    CONTINUE                                                             EVA01520
      CLI=CLI*CSTI                                                         EVA01530
      AF=AF*CSTA                                                           EVA01540
26    CALL NONLIN(CLIV(NCLI(1)),CLIV(NCLI(2)),CLIV(NCLI(3)),CT(1),CT(2),EVA01550
     1CT(3),CLI,CTTO1)                                                     EVA01560
      CALL NONLIN(CLIV(NCLI(1)),CLIV(NCLI(2)),CLIV(NCLI(3)),CT(4),CT(5),EVA01570
     1CT(6),CLI,CTTO2)                                                     EVA01580
      CALL NONLIN(CLIV(NCLI(1)),CLIV(NCLI(2)),CLIV(NCLI(3)),CT(7),CT(8),EVA01590
     1CT(9),CLI,CTTO3)                                                     EVA01600
31    CALL NONLIN(AFVST(NAF(1)),AFVST(NAF(2)),AFVST(NAF(3)),CTTO1,         EVA01610
     1CTTO2,CTTO3,AF,CTTO)                                                 EVA01620
      CLI=CLI/CSTI                                                         EVA01630
      AF=AF/CSTA                                                           EVA01640
28    IF(CTTO.LE.0.)GO TO 33                                               EVA01650
      TSTAT=FLOAT(NUMENG)*(((CTTO*BHPTO*33000.)/((RPMTO/(GR*CSTG))*D))+    EVA01660
     1TJET   )                                                             EVA01670
121   DO 43 I=1,J1                                                         EVA01680
      IF((AF*CSTA).LT.AFVFLT(1))GO TO 44                                   EVA01690
      IF((AF*CSTA).GE.AFVFLT(I).AND.(AF*CSTA).LE.AFVFLT(I+1))GO TO 44      EVA01700
43    CONTINUE                                                             EVA01710
```

```
   44 IF(I.EQ.J1)GO TO 45                                               EVA01720
      NAF(1)=I                                                          EVA01730
      NAF(2)=I+1                                                        EVA01740
      NAF(3)=I+2                                                        EVA01750
      GO TO 46                                                          EVA01760
   45 NAF(1)=I+1                                                        EVA01770
      NAF(2)=I                                                          EVA01780
      NAF(3)=I-1                                                        EVA01790
   46 IF(MM.EQ.0.AND.VOITO.EQ.0..AND.TOCONT.EQ.0.)GO TO 80              EVA01800
      VJ=(101.4*VTOKTS*GR*CSTG)/(RPMTO*D)                               EVA01810
  110 CALL BLDANG(CPTO,0.,BETAST)                                       EVA01820
      IF(NTYPE.EQ.1)GO TO 75                                            EVA01830
      CALL NONLIN(0.,100.,400.,BHPTO1,BHPTO2,BHPTO3,VTOKTS,BHP1)        EVA01840
      BHPTO=BHP1+ALTCOR                                                 EVA01850
      CALL LIN(0.,100.,TJETM1,TJETM2,VTOKTS,TJET   )                    EVA01860
      IF(BHPTO.GT.BHPLIM)TJET=(1.-(BHPTO-BHPLIM)/BHPTO)*TJET            EVA01870
      IF(BHPTO.GT.BHPLIM)BHPTO=BHPLIM                                   EVA01880
      CPTO=BHPTO/(2000.*SIGMA(ALTTO)*((RPM/(1000.*GR*CSTG))**3)*((D/10.)EVA01890
     1**5))                                                             EVA01900
   75 CALL ETA(CPTO,VJ,ETALO)                                           EVA01910
      CALL COMP(VJ,VTOKTS,ALTTO,FSUBT,FSUBB)                            EVA01920
      FSUBB=FSUBB/.94                                                   EVA01930
      ETALO=ETALO*FSUBT*FSUBB                                           EVA01940
      TLO=FLOAT(NUMENG)*((326.*BHPTO*ETALO/VTOKTS)+TJET   )             EVA01950
      A=TSTAT-FCOEF*WT                                                  EVA01960
      B=(TLO-TSTAT)/((VTOKTS*1.689)**2)                                 EVA01970
      C=.5*RHOSL*WAREA*SIGMA(ALTTO)*(FCOEF*CLOPT-CD)                    EVA01980
      Z=B+C                                                             EVA01990
      Y=((VTOKTS*1.689)**2+(A/Z))/(A/Z)                                 EVA02000
      IF(Y.LE.0.)GO TO 33                                               EVA02010
      TODIST=(WT*ALOG(Y))/(2.*G*Z)                                      EVA02020
      IF(TODIST.GT.TOCONT.AND.TOCONT.NE.0.)TOFAC=1.                     EVA02030
      IF(TODIST.LT.0.)GO TO 33                                          EVA02040
      IF((GRS*DS*AFS*CLIS*FLOAT(NUMBLS)).NE.0..OR.NN.EQ.1)              EVA02050
     1WRITE(6,1097)TODIST,CPTO1,CTTO,ETALO,BETAST,CPTO,VJ,EFFM,FSUBT,   EVA02060
     2FSUBB,BHPTO,TJET                                                  EVA02070
      GO TO 80                                                          EVA02080
   33 TOLT=1.                                                           EVA02090
      IF((GRS*DS*AFS*CLIS*FLOAT(NUMBLS)).EQ.0..AND.NN.EQ.0)GO TO 100    EVA02100
      WRITE(6,2000)CPTO1,CTTO                                           EVA02110
      GO TO 92                                                          EVA02120
 2000 FORMAT(/' CP ='F7.4,//' CT/CP ='F6.3,//' AIRCRAFT WILL NOT TAKEOFFEVA02130
     1 UNDER SPECIFIED CONDITIONS')                                     EVA02140
 1097 FORMAT(//' TAKEOFF DISTANCE ='F7.0,' FT'/' CP (STATIC) ='F7.4,/   EVA02150
     1' CT/CP ='F6.3,/' PROP EFFICIENCY (AT LIFTOFF) = 'F4.3,/          EVA02160
     2' BLADE ANGLE (.75R) (AT V=0.) = 'F4.1,' DEG'//' FOLLOWING VALUES EVA02170
     3ARE CALCULATED AT LIFTOFF :'/2X,'CP',7X,'J',7X,'M(EFF)',6X,'FT',8XEVA02180
     4,'FB',8X,'BHP',6X,'TJET'/F5.3,4F10.4,F10.0,F9.0)                  EVA02190
```

```
   80 IF(MM.EQ.0.AND.VOIVM.EQ.0..AND.VMCONT.EQ.0.)GO TO 101               EVA02200
      DELTA=.05                                                         EVA02210
      N1=0                                                              EVA02220
      N2=0                                                              EVA02230
      X=.25                                                             EVA02240
      VKTS=AVM                                                          EVA02250
   76 IF(VKTS.LE.0.)X=X+1.                                              EVA02260
      IF(VKTS.LE.0.)VKTS=AVM*X                                          EVA02270
      IF(VKTS.GE.550.)GO TO 85                                          EVA02280
      IF(NTYPE.EQ.1)GO TO 67                                            EVA02290
      RPMVM=RPM                                                         EVA02300
      CALL NONLIN(0.,100.,400.,BHPTO1,BHPTO2,BHPTO3,VKTS,BHP1)          EVA02310
      CALL LIN(0.,10000.,BHP2,BHPM,ALTVM,BHP3)                          EVA02320
      ALTCOR=BHP3-BHP2                                                  EVA02330
      BHPVM=BHP1+ALTCOR                                                 EVA02340
      CALL LIN(0.,100.,TJETM1,TJETM2,VKTS,TJET)                         EVA02350
      IF(BHPVM.GT.BHPLIM)TJET=(1.-(BHPVM-BHPLIM)/BHPVM)*TJET            EVA02360
      IF(BHPVM.GT.BHPLIM)BHPVM=BHPLIM                                   EVA02370
   67 CPVM=BHPVM/(2000.*SIGMA(ALTVM)*((RPMVM/(1000.*GR*CSTG))**3)*((D/10EVA02380
     1.)**5))                                                           EVA02390
      CLVM=(2.*WT)/(RHOSL*SIGMA(ALTVM)*WAREA*((VKTS*1.689)**2))         EVA02400
      CDVM=CDPCR+((CLVM**2)/(3.141593*AR*E))                            EVA02410
      REQHP=.5*RHOSL*SIGMA(ALTVM)*WAREA*CDVM*((VKTS*1.689)**3)/550.     EVA02420
      VJ=(101.4*VKTS*GR*CSTG)/(RPMVM*D)                                 EVA02430
  111 CALL ETA(CPVM,VJ,ETAVM)                                           EVA02440
      CALL COMP(VJ,VKTS,ALTVM,FSUBT,FSUBB)                              EVA02450
      ETAVM=ETAVM*FSUBT*FSUBB                                           EVA02460
      AVHP=((TJET*VKTS*1.689/550.)+BHPVM*ETAVM)*FLOAT(NUMENG)           EVA02470
      IF((DELTA*AVM).LE..5.OR.ABS(AVHP-REQHP).LE.1.)GO TO 65            EVA02480
      IF(AVHP.GT.REQHP)GO TO 66                                         EVA02490
      IF(N1.EQ.1)DELTA=DELTA*.6182                                      EVA02500
      N2=1                                                              EVA02510
      VKTS=VKTS-DELTA*AVM                                               EVA02520
      GO TO 76                                                          EVA02530
   66 IF(N2.EQ.1)DELTA=DELTA*.6182                                      EVA02540
      N1=1                                                              EVA02550
      VKTS=VKTS+DELTA*AVM                                               EVA02560
      GO TO 76                                                          EVA02570
   65 VMAX=VKTS+(AVHP-REQHP)/(AVHP/VKTS)                                EVA02580
      CALL BLDANG(CPVM,VJ,BETAVM)                                       EVA02590
      IF(NTYPE.EQ.2)BHPTO=BHPLIM                                        EVA02600
      IF(VMCONT.NE.0..AND.VMAX.LT.VMCONT)VMFAC=1.                       EVA02610
      IF((GRS*DS*AFS*CLIS*FLOAT(NUMBLS)).NE.0..OR.NN.EQ.1)              EVA02620
     1WRITE(6,2001)VMAX,BETAVM,ETAVM,CPVM,VJ,EFFM,FSUBT,FSUBB,BHPVM,TJETEVA02630
 2001 FORMAT(//' MAXIMUM SPEED = 'F5.1,' KTS'/' BLADE ANGLE (.75R) = '  EVA02640
     1F4.1,' DEG'/' PROP EFFICIENCY = 'F4.3//2X,'CP',7X,'J',7X,'M(EFF)',EVA02650
     26X,'FT',8X,'FB',8X,'BHP',6X,'TJET'/F5.3,4F10.4,F10.0,F9.0)        EVA02660
  101 IF(MM.EQ.0.AND.VOICR.EQ.0..AND.VOIRA.EQ.0..AND.VOIEN.EQ.0..AND.   EVA02670
```

```
     1CRCONT.EQ.0..AND.RACONT.EQ.0..AND.ENCONT.EQ.0.)GO TO 122          EVA02680
      DELTA=.05                                                         EVA02690
      N1=0                                                              EVA02700
      N2=0                                                              EVA02710
      X=.25                                                             EVA02720
      VKTS=AVCR                                                         EVA02730
   77 IF(VKTS.LE.0.)X=X+1.                                              EVA02740
      IF(VKTS.LE.0.)VKTS=AVCR*X                                         EVA02750
      IF(VKTS.GE.550.)GO TO 83                                          EVA02760
      IF(NTYPE .EQ.1)GO TO 68                                           EVA02770
      RPMCR=RPM                                                         EVA02780
      AVCR1=AVCR+50.                                                    EVA02790
      CALL LIN(AVCR,AVCR1,BHPCR1,BHPCR2,VKTS,BHPCR)                     EVA02800
      CALL LIN(C.,100.,TJETC1,TJETC2,VKTS,TJET)                         EVA02810
      IF(BHPCR.GT.BHPLIM)TJET=(1.-(BHPCR-BHPLIM)/BHPCR)*TJET            EVA02820
      IF(BHPCR.GT.BHPLIM)BHPCR=BHPLIM                                   EVA02830
   68 CPCR=BHPCR/(2000.*SIGMA(ALTCR)*((RPMCR/(1000.*GR*CSTG))**3)*((D/10EVA02840
     1.)**5))                                                           EVA02850
      CLCR=(2.*WT)/(RHOSL*SIGMA(ALTCR)*WAREA*((VKTS*1.689)**2))         EVA02860
      CDCR=CDPCR+((CLCR**2)/(3.141593*AR*E))                            EVA02870
      REQHP=.5*RHOSL*SIGMA(ALTCR)*WAREA*CDCR*((VKTS*1.689)**3)/550.     EVA02880
      VJ=(101.4*VKTS*GR*CSTG)/(RPMCR*D)                                 EVA02890
  112 CALL ETA(CPCR,VJ,ETACR)                                           EVA02900
      CALL COMP(VJ,VKTS,ALTCR,FSUBT,FSUBB)                              EVA02910
      ETACR=ETACR*FSUBT*FSUBB                                           EVA02920
      AVHP=((TJET*VKTS*1.689/550.)+BHPCR*ETACR)*FLOAT(NUMENG)           EVA02930
      IF((DELTA*AVCR).LE..5.OR.ABS(AVHP-REQHP).LE.1.)GO TO 69           EVA02940
      IF(AVHP.GT.REQHP)GO TO 70                                         EVA02950
      IF(N1.EQ.1)DELTA=DELTA*.6182                                      EVA02960
      N2=1                                                              EVA02970
      VKTS=VKTS-DELTA*AVCR                                              EVA02980
      GO TO 77                                                          EVA02990
   70 IF(N2.EQ.1)DELTA=DELTA*.6182                                      EVA03000
      N1=1                                                              EVA03010
      VKTS=VKTS+DELTA*AVCR                                              EVA03020
      GO TO 77                                                          EVA03030
   69 VCRKTS=VKTS+(AVHP-REQHP)/(AVHP/VKTS)                              EVA03040
      CALL BLDANG(CPCR,VJ,BETACR)                                       EVA03050
      IF(CRCONT.NE.0..AND.VCRKTS.LT.CRCONT)CRFAC=1.                     EVA03060
      IF((GRS*DS*AFS*CLIS*FLOAT(NUMBLS)).NE.0..OR.NN.EQ.1)              EVA03070
     1WRITE(6,2002)VCRKTS,BETACR,ETACR,CPCR,VJ,EFFM,FSUBT,FSUBB,BHPCR,  EVA03080
     2TJET                                                              EVA03090
 2002 FORMAT(//' CRUISE SPEED = 'F5.1,' KTS'/' BLADE ANGLE (.75R) = ',  EVA03100
     1F4.1,' DEG'/' PROP EFFICIENCY = 'F4.3//2X,'CP',7X,'J',7X,'M(EFF)',EVA03110
     26X,'FT',8X,'FB',8X,'BHP',6X,'TJET'/F5.3,4F10.4,F10.0,F9.0)        EVA03120
      IF(NTYPE.EQ.1)GO TO 90                                            EVA03130
      CALL LIN(AVCR,AVCR1,FCON1,FCON2,VCRKTS,FCON)                      EVA03140
   90 ENDUR=FUEL/(FCON*FLOAT(NUMENG))-.75                               EVA03150
```

```
      IF(ENCONT.NE.0..AND.ENDUR.LT.ENCONT)ENFAC=1.                      EVA03160
      DELV=FLOAT(NUMENG)*2.*(WT-FUEL/2.)*FCON*ENDUR/(RHOSL*SIGMA(ALTCR)*EVA03170
     1WAREA*3.1416*AR*E*(RHOSL*SIGMA(ALTCR)*WAREA*CDPCR*((1.689*VCRKTS)*EVA03180
     2*3)-275.*AVHP))                                                   EVA03190
      V=(2.*VCRKTS+DELV/1.689)/2.                                       EVA03200
      RANGE=V*ENDUR                                                     EVA03210
      IF(RACONT.NE.0..AND.RANGE.LT.RACONT)RAFAC=1.                      EVA03220
      IF((GPS*DS*AFS*CLIS*FLOAT(NUMBLS)).NE.0..OR.NN.EQ.1)              EVA03230
     1WRITE(6,2003)RANGE,ENDUR                                          EVA03240
 2003 FORMAT(//' RANGE (AT CRUISE CONDITIONS AND CRUISE SPEED) = 'F6.0, EVA03250
     1' N MI  (45 MIN RESERVE)'//' ENDURANCE (AT CRUISE POWER AND CONDITEVA03260
     2IONS) = 'F5.1,' HRS  (45 MIN RESERVE)')                           EVA03270
  122 IF(MM.EQ.0.AND.VOICL.EQ.0..AND.CLCONT.EQ.0..AND.TOLT.EQ.0..AND.   EVA03280
     1VMLT.EQ.0..AND.VCRLT.EQ.0.)GO TO 123                              EVA03290
  100 DELHP1=0.                                                         EVA03300
      N1=0                                                              EVA03310
      DELTA=.005                                                        EVA03320
      VKTS=AVCL                                                         EVA03330
   78 IF(NTYPE.EQ.1)GO TO 71                                            EVA03340
      RPMCL=RPM                                                         EVA03350
      CALL NONLIN(0.,100.,400.,BHPT01,BHPT02,BHPT03,VKTS,BHPCL1)        EVA03360
      CALL LIN(10000.,0.,BHPM,BHP2,ALTCL,BHP3)                          EVA03370
      ALTCOR=BHP3-BHP2                                                  EVA03380
      BHPCL=BHPCL1+ALTCOR                                               EVA03390
      CALL LIN(0.,100.,TJETM1,TJETM2,VKTS, TJET)                        EVA03400
      IF(BHPCL.GT.BHPLIM)TJET=(1.-(BHPCL-BHPLIM)/BHPCL)*TJET            EVA03410
      IF(BHPCL.GT.BHPLIM)BHPCL=BHPLIM                                   EVA03420
   71 CPCL=BHPCL/(2000.*SIGMA(ALTCL)*((RPMCL/(1000.*GR*CSTG))**3)*((D/10EVA03430
     1.)**5))                                                           EVA03440
      CLCL=(2.*WT)/(RHOSL*SIGMA(ALTCL)*WAREA*((VKTS*1.689)**2))         EVA03450
      CDCL=CDPCR+((CLCL**2)/(3.141593*AR*E))                            EVA03460
      REQHP=.5*RHOSL*SIGMA(ALTCL)*WAREA*CDCL*((VKTS*1.689)**3)/550.     EVA03470
      VJ=(101.4*VKTS*GR*CSTG)/(RPMCL*D)                                 EVA03480
  113 CALL ETA(CPCL,VJ,ETACL)                                           EVA03490
      CALL COMP(VJ,VKTS,ALTCL,FSUBT,FSUBB)                              EVA03500
      ETACL=ETACL*FSUBT*FSUBB                                           EVA03510
      AVHP=((TJET*VKTS*1.689/550.)+BHPCL*ETACL)*FLOAT(NUMENG)           EVA03520
      DELHP2=DELHP1                                                     EVA03530
      DELHP1=AVHP-REQHP                                                 EVA03540
      IF(DELHP1.LT.DELHP2)GO TO 72                                      EVA03550
      N1=N1+1                                                           EVA03560
      GO TO 73                                                          EVA03570
   72 IF(N1.GT.1)GO TO 74                                               EVA03580
      DELTA=-DELTA                                                      EVA03590
   73 VKTS=VKTS+DELTA*VKTS                                              EVA03600
      GO TO 78                                                          EVA03610
   74 RC=DELHP2*33000./WT                                               EVA03620
      CALL BLDANG(CPCL,VJ,BETACL)                                       EVA03630
```

```
      IF(CLCONT.NE.0..AND.RC.LT.CLCONT)CLFAC=1.                         EVA03640
      VCLKTS=VKTS                                                       EVA03650
      IF((GRS*DS*AFS*CLIS*FLCAT(NUMBLS)).NE.0..OR.NN.EQ.1)              EVA03660
     1WRITE(6,2004)RC,VCLKTS,BETACL,ETACL,      CPCL,VJ,EFFM,FSUBT,FSUBBEVA03670
     2,BHPCL,TJET                                                       EVA03680
 2004 FORMAT(//' MAXIMUM RATE OF CLIMB ='F6.0,' FT/MIN'/' CLIMB SPEED ='EVA03690
     1F6.0,' KTS'/' BLADE ANGLE (.75R) = 'F4.1,' DEG'/' PROP EFFICIENCY EVA03700
     2= 'F4.3//2X,'CP',7X,'J',7X,'M(EFF)',6X,'FT',8X,'FB',8X,'BHP',6X,  EVA03710
     3'TJET'/F5.3,4F10.4,F10.0,F9.0)                                    EVA03720
  123 IF(TOLT.EQ.1..OR.VMLT.EQ.1..OR.VCRLT.EQ.1.)GO TO 102              EVA03730
      GO TO 91                                                          EVA03740
   83 VCRLT=1.                                                          EVA03750
      IF((GRS*DS*AFS*CLIS*FLCAT(NUMBLS)).EQ.0..AND.NN.EQ.0)GO TO 100    EVA03760
      WRITE(6,2051)                                                     EVA03770
      GO TO 100                                                         EVA03780
 2051 FORMAT(/' AIRCRAFT WILL NOT FLY AT SPECIFIED ENGINE CRUISE POWER AEVA03790
     1ND CONDITIONS')                                                   EVA03800
   85 VMLT=1.                                                           EVA03810
      IF((GRS*DS*AFS*CLIS*FLCAT(NUMBLS)).EQ.0..AND.NN.EQ.0)GO TO 100    EVA03820
      WRITE(6,2050)                                                     EVA03830
      GO TO 101                                                         EVA03840
 2050 FORMAT(/' AIRCRAFT WILL NOT FLY AT SPECIFIED ENGINE MAX. SPEED POWEVA03850
     1ER AND CONDITIONS')                                               EVA03860
   91 IF(VMAX.EQ.0.)VMAX=1.                                             EVA03870
      IF(VCRKTS.NE.0.)GO TO 124                                         EVA03880
      VCRKTS=1.                                                         EVA03890
      RANGE=1.                                                          EVA03900
      ENDUR=1.                                                          EVA03910
  124 IF(RC.EQ.0.)RC=1.                                                 EVA03920
      IF((GRS*DS*AFS*CLIS*FLOAT(NUMBLS)).NE.0.)GO TO 92                 EVA03930
      FUNCT=(VOITO*TODIST/ATO)+(VOIVM*AVM/VMAX)+(VOICR*AVCR/VCRKTS)+    EVA03940
     1(VOIEN*AENDUR/ENDUR)+(VOIRA*ARANGE/RANGE)+(VOICL*ARC/RC)+         EVA03950
     2(TOFAC*1.E-01*TODIST)+(VMFAC*1.E06/VMAX)+(CRFAC*1.E07/VCRKTS)+    EVA03960
     3(RAFAC*1.E09/RANGE)+(ENFAC*1.E07/ENDUR)+(CLFAC*1.E10/RC)          EVA03970
  102 IF(TOLT.EQ.1..OR.VMLT.EQ.1..OR.VCRLT.EQ.1.)FUNCT=10000.-RC        EVA03980
      IF((TOFAC.NE.1..AND.CLFAC.NE.1..AND.CRFAC.NE.1..AND.VMFAC.NE.1.   EVA03990
     1.AND.ENFAC.NE.1..AND.RAFAC.NE.1.).OR.(MM.EQ.1.AND.NN.EQ.0).OR.(MM.EVA04000
     2EQ.0.AND.NN.EQ.0))GO TO 92                                        EVA04010
      IF((GRS*DS*AFS*CLIS*FLOAT(NUMBLS)).NE.0..OR.NN.EQ.1)WRITE(6,2052) EVA04020
 2052 FORMAT(/' AIRCRAFT CANNOT OPERATE WITHIN SPECIFIED PERFORMANCE CONEVA04030
     1STRAINTS')                                                        EVA04040
   92 RETURN                                                            EVA04050
      END                                                               EVA04060
```

# APPENDIX B

## SAMPLE COMPUTER OUTPUT

```
EXECUTION BEGINS...
DATA NOW BEING READ
DATA HAS BEEN READ
THIS PROGRAM OPTIMIZES THE PROPELLER FOR A GIVEN AIRCRAFT UNDER SPECIFIED CONDITIONS

PLEASE TYPE IN THE REQUESTED INFORMATION, FOLLOWING THE DIRECTIONS CAREFULLY

INPUT ROUTINE...
        STEP 1:  AIRCRAFT DATA
        STEP 2:  ENGINE DATA
        STEP 3:  AIRCRAFT ENVIRONMENT AND ANTICIPATED PERFORMANCE
        STEP 4:  AIRCRAFT PERFORMANCE CONSTRAINTS  (OPTIMIZATION ONLY)
        STEP 5:  PROPELLER CONSTRAINTS
        STEP 6:  OPTIMIZATION CRITERIA  (OPTIMIZATION ONLY)
        STEP 7:  END OF INPUT ROUTINE...PROPELLER EVALUATION OR OPTIMIZATION


STEP 1: AIRCRAFT DATA
PLEASE TYPE IN AIRCRAFT DATA AS CALLED FOR IN THE PROPER UNITS....INCLUDE DECIMAL

DESIGN GROSS WEIGHT = ?   (LBS.)
105000.

WING AREA = ?    (SQ. FT.)
1300.

NUMBER OF ENGINES = ?
4.

ASPECT RATIO = ?
7.5

PARASITIC DRAG COEFFICIENT = ? (TAKEOFF CONFIGURATION)
.05

PARASITIC DRAG COEFFICIENT = ?  (CRUISE CONFIGURATION)
.0247
```

```
WING EFFICIENCY FACTOR = ?  (SMALL E)
.98

FUEL CAPACITY = ?   (LBS.)
29370.

TAKEOFF SPEED = ?    (KTS TAS)
121.


END OF STEP 1....AIRCRAFT DATA FOLLOW:        (CHECK VALUES CAREFULLY FOR ACCURACY)

WEIGHT =  105000.0 LBS.                WING AREA =    1300.0 SQ. FT.

NUMBER OF ENGINES = 4                  ASPECT RATIO =      7.50

PARASITIC DRAG COEF. (TAKEOFF) = 0.050 PARASITIC DRAG COEF. (CRUISE) = 0.025

WING EFFICIENCY FACTOR =     0.980     FUEL CAPACITY =   29370.0 LBS.

TAKEOFF SPEED =      121. KTS TAS


AT THIS POINT, TYPE "0." IF YOU WISH TO CONTINUE
IF YOU WISH TO RETURN OR ADVANCE TO ANOTHER STEP, TYPE NUMBER OF STEP
0.


STEP 2: ENGINE DATA
PLEASE TYPE IN REQUIRED ENGINE DATA IN PROPER UNITS....ALWAYS INCLUDE A DECIMAL
ENTER ENGINE CHARACTERISTICS AT WHATEVER AIR DENSITY THE AIRCRAFT PERFORMANCE IS TO BE OPTIMIZED,
FOR THAT PARTICULAR PORTION OF ITS MISSION

RECIPROCATING (1) OR TURBO-PROP (2) ENGINE ?....(TYPE NUMBER CORRESPONDING TO CORRECT CHOICE)
2.


ENGINE OPERATING RPM = ?
13820.
```

```
SHP (PER ENGINE) AT CRUISE ALTITUDE AND CRUISE POWER SETTING,
FOR ANTICIPATED CRUISE SPEED (TAS) AND 50 KTS ABOVE THIS SPEED, RESPECTIVELY, = ? (HIT RETURN BETWEEN ENTRIES)
1755.
1861.

ENTER SHP AT MAX. POWER SETTING (POWER FOR TAKEOFF AND MAX. SPEED)
FOR ANTICIPATED MAX. SPEED (TAS) AND 10,000 FT MSL ALTITUDE
SHP AT ABOVE CONDITIONS = ?
4637.

ENTER SEA LEVEL SHP FOR MAX. POWER SETTING AT 0., 100, AND 400 KTS (TAS), RESPECTIVELY
SHP UNDER ABOVE CONDITIONS = ? (HIT RETURN AFTER EACH SHP ENTRY)
4527.
4712.
5790.

SEA LEVEL JET THRUST (LBS PER ENGINE) FOR MAX. POWER SETTING AT 0. AND 100. KTS (TAS), RESPECTIVELY, = ?
(HIT RETURN AFTER EACH ENTRY)
639.
533.

SEA LEVEL JET THRUST FOR CRUISE POWER SETTING AT 0. AND 100. KTS (TAS), RESPECTIVELY, = ? (HIT RETURN AFTER EACH ENTRY
462.
353.

ENTER FUEL FLOW (LBS/HR) FOR CRUISE POWER AND ALTITUDE AT ANTICIPATED CRUISE SPEED (TAS)
AND 50 KTS ABOVE THIS SPEED, RESPECTIVELY
FUEL FLOW AT ABOVE CONDITIONS = ? (HIT RETURN AFTER EACH ENTRY)
1035.
1063.

LIMITING SHP FOR CONTINUOUS OPERATION = ?
4500.
```

```
END OF STEP 2....ENGINE DATA (PER ENGINE) FOLLOW       (CHECK VALUFS CAREFULLY FOR ACCURACY)

ENGINE OPERATING RPM =   13820.0

SHP AT CRUISE POWER AND ALTITUDE =    1755.0 AT CRUISE SPD AND =    1861.0 AT CRUISE SPD + 50 KTS

SHP AT MAX. SPEED POWER AND 10,000 FT ALTITUDE FOR ANTICIPATED MAX. SPEED =    4637.0

SEA LEVEL SHP AT MAX. POWER =    4527.0 AT 0. KTS,    4712.0 AT 100 KTS,    5790.0 AT 400 KTS

SEA LEVEL JET THRUST FOR MAX. POWER =    639.0 LBS AT 0. KTS,   533.0 LBS AT 100 KTS

SEA LEVEL JET THRUST FOR CRUISE POWER =   462.0 LBS AT 0. KTS,   353.0 LBS AT 100 KTS

FUEL FLOW FOR CRUISE POWER AND ALTITUDE =  1035.0 LBS/HR AT CRUISE SPEED,  1063.0 LBS/HR AT CRUISE SPEED + 50 KTS

LIMITING SHP FOR CONTINUOUS OPERATION =    4500.0


AT THIS POINT, TYPE "0." IF YOU WISH TO CONTINUE
IF YOU WISH TO RETURN OR ADVANCE TO ANOTHER STEP, TYPE NUMBER OF STEP
0.


STEP 3:  AIRCRAFT ENVIRONMENT AND ANTICIPATED PERFORMANCE
PLEASE TYPE IN REQUIRED DATA IN PROPER UNITS....ALWAYS INCLUDE A DECIMAL

PLEASE TYPE IN ALTITUDES (FT. MSL) AT WHICH AIRCRAFT PERFORMANCE IS TO BE EVALUATED
AT TAKEOFF, CLIMB,CRUISE, AND MAXIMUM SPEED....(SHOULD CORRESPOND WITH ENGINE DATA)

ALTITUDE AT TAKEOFF = ?  (FT. MSL)
0.

ALTITUDE AT CLIMB = ?
0.

CRUISE ALTITUDE = ?
15000.
```

```
ALTITUDE AT MAX. SPEED = ?
10000.1

ANTICIPATED AIRCRAFT SPEEDS (TAS):

CLIMB SPEED = ?  (KTS)
235.

CRUISE SPEED = ?  (KTS)
250.

MAX. SPEED = ?  (KTS)
386.


END OF STEP 3....ENVIRONMENT AND ANTICIPATED SPEED DATA FOLLOW  (CHECK VALUES CAREFULLY FOR ACCURACY)

ALTITUDE AT TAKEOFF =       0.0 FT MSL           ALTITUDE AT CLIMB =       0.0 FT MSL

CRUISE ALTITUDE =   15000.0 FT MSL               ALTITUDE AT MAX. SPEED =   10000.1 FT MSL

CLIMB SPEED =  235.0 KTS (TAS)       CRUISE SPEED =  250.0 KTS (TAS)        MAX. SPEED =  386.0 KTS (TAS)


AT THIS POINT, TYPE "0." IF YOU WISH TO CONTINUE
IF YOU WISH TO RETURN OR ADVANCE TO ANOTHER STEP, TYPE NUMBER OF STEP
0.


STEP 4:  PERFORMANCE CONSTRAINTS
PLEASE TYPE IN REQUIRED DATA IN PROPER UNITS....ALWAYS INCLUDE A DECIMAL

TYPE "0." AS THE VALUE OF ANY CONSTRAINT WHICH YOU DO NOT WISH TO SPECIFY

MAXIMUM ALLOWABLE TAKEOFF DISTANCE = ?  (FT)
2500.
```

```
LOWEST ALLOWABLE MAXIMUM RATE OF CLIMB = ?  (FT/MIN)
0.

MINIMUM ALLOWABLE RANGE AT CRUISE CONDITIONS = ?(N MI)
0.

MINIMUM ALLOWABLE ENDURANCE AT CRUISE CONDITIONS = ?  (HRS)
0.

MINIMUM ALLOWABLE CRUISE SPEED = ?  (KTS TAS)
0.

LOWEST ALLOWABLE MAXIMUM SPEED = ?  (KTS TAS)
0.


END OF STEP 4....PERFORMANCE CONSTRAINTS FOLLOW   (CHECK VALUES CAREFULLY FOR ACCURACY)

MAX. ALLOWABLE TAKEOFF DISTANCE =  2500.0 FT/MIN
LOWEST ALLOWABLE MAX. RATE OF CLIMB =    0.0 FT/MIN ....(UNSPECIFIED)

MINIMUM ALLOWABLE RANGE AT CRUISE CONDITIONS =     0.0 N MI ....(UNSPECIFIED)

MINIMUM ALLOWABLE ENDURANCE AT CRUISE CONDITIONS =  0.0 HRS ....(UNSPECIFIED)

MINIMUM ALLOWABLE CRUISE SPEED =   0.0 KTS ....(UNSPECIFIED)

LOWEST ALLOWABLE MAX. SPEED =   0.0 KTS ....(UNSPECIFIED)


AT THIS POINT, TYPE "0." IF YOU WISH TO CONTINUE
IF YOU WISH TO RETURN OR ADVANCE TO ANOTHER STEP, TYPE NUMBER OF STEP
0.0.@@


STEP 5:  PROPELLER CONSTRAINTS
PLEASE TYPE IN REQUIRED DATA IN PROPER UNITS....ALWAYS INCLUDE A DECIMAL

MAX. BLADE DIAMETER FOR STRUCTURAL OR GROUND CLEARANCE = ?  (FT)
13.5
```

```
MAX. BLADE DIAMETER = 13.50 FT

THE FOLLOWING PROP CONSTRAINTS ARE OPTIONAL
IF NOT SPECIFIED, THE OPTIMUM VALUE OF EACH QUANTITY WILL BE DETERMINED
TYPE "0." AS THE VALUE OF ANY CONSTRAINT WHICH YOU DO NOT WISH TO SPECIFY

NUMBER OF BLADES = ?
0.

GEAR RATIO = ?  (ENGINE RPM/PROP RPM)
13.54

BLADE DIAMETER = ?  (FT)
0.

ACTIVITY FACTOR = ?
0.

INTEGRATED LIFT COEFFICIENT = ?
0.


END OF STEP 5....OPTIONAL BLADE CONSTRAINTS FOLLOW     (CHECK VALUES CAREFULLY FOR ACCURACY)

GEAR RATIO =  13.54  (ENGINE RPM/PROP RPM)
NUMBER OF BLADES = 0 ....(UNSPECIFIED)

BLADE DIAMETER =  0.0  FT ....(UNSPECIFIED)

ACTIVITY FACTOR =    0.0 ....(UNSPECIFIED)

INTEGRATED LIFT COEFFICIENT = 0.0 ....(UNSPECIFIED)


AT THIS POINT, TYPE "0." IF YOU WISH TO CONTINUE
IF YOU WISH TO RETURN OR ADVANCE TO ANOTHER STEP, TYPE NUMBER OF STEP
0.
```

```
STEP 6:  OPTIMIZATION CRITERIA
PLEASE RATE EACH OF THE FOLLOWING PERFORMANCE TRAITS, ON A SCALE FROM 0. TO 10.,
ACCORDING TO RELATIVE IMPORTANCE TO AIRCRAFT MISSION

VALUES ASSIGNED SHOULD BE RELATIVE ONLY TO THE OTHER VALUES
ABSOLUTE MAGNITUDE IS ARBITRARY

"VALUE OF IMPORTANCE" OF SHORT TAKEOFF DISTANCE = ?
0.

V.O.I. OF CLIMB RATE = ?
0.

V.O.I. OF RANGE = ?
0.

V.O.I. OF ENDURANCE = ?
0.

V.O.I. OF CRUISE SPEED = ?
1.

V.O.I. OF MAX. SPEED = ?
0.


END OF STEP 6....OPTIMIZATION CRITERIA FOLLOW          (CHECK VALUES CAREFULLY FOR ACCURACY)

V.O.I. OF SHORT TAKEOFF DISTANCE =   0.0
V.O.I. OF CLIMB RATE =   0.0
V.O.I. OF RANGE =   0.0
V.O.I. OF ENDURANCE =   0.0
V.O.I. OF CRUISE SPEED =   1.0
V.O.I. OF MAX. SPEED =   0.0


AT THIS POINT, TYPE "0." IF YOU WISH TO CONTINUE
IF YOU WISH TO RETURN OR ADVANCE TO ANOTHER STEP, TYPE NUMBER OF STEP
0.
```

```
END OF INPUT ROUTINE....YOU SHOULD HAVE TIME FOR A QUICK CUP OF COFFEE WHILE AWAITING RESULTS

BEGIN 4 BLADE SEARCH

DIRECT SEARCH  DELCAP =   0.816000E 00  RHO =   0.250000E 00  DELLC =   0.500000E-01  MAXEVL =     500   KN = -1

 MOVE    FUNCTION VALUE        GEAR RATIO        PROP DIAM.     ACTIVITY FACTOR    INTEG. LIFT COEF.

ORIGIN    0.8650219E 00          13.54             13.50            220.0              0.400

EXPLORE   0.8631015E 00          13.54             13.50            220.0              0.318
PATTERN   0.8631015E 00          13.54             13.50            220.0              0.318

EXPLORE   0.8631015E 00          13.54             13.50            220.0              0.318

EXPLORE   0.8631015E 00          13.54             13.50            220.0              0.318

EXPLORE   0.8630986E 00          13.54             13.50            220.0              0.323
PATTERN   0.8630986E 00          13.54             13.50            220.0              0.323

EXPLORE   0.8630986E 00          13.54             13.50            220.0              0.323

EXPLORE   0.8630978E 00          13.54             13.50            220.0              0.322
PATTERN   0.8630978E 00          13.54             13.50            220.0              0.321

EXPLORE   0.8630971E 00          13.54             13.50            220.0              0.321
PATTERN   0.8630971E 00          13.54             13.50            220.0              0.321

EXPLORE   0.8630971E 00          13.54             13.50            220.0              0.321

END OF 4 BLADE SEARCH...         75 FUNCTION EVALUATIONS PERFORMED

BEGIN 3 BLADE SEARCH

DIRECT SEARCH  DELCAP =   0.816000E 00  RHO =   0.250000E 00  DELLC =   0.500000E-01  MAXEVL =     500   KN = -1

 MOVE    FUNCTION VALUE        GEAR RATIO        PROP DIAM.     ACTIVITY FACTOR    INTEG. LIFT COEF.

ORIGIN    0.2562007E 03          13.54             13.50            220.0              0.400


EXPLORE   0.2517228E 03          13.54             13.50            220.0              0.482
PATTERN   0.8662022E 00          13.54             13.50            220.0              0.563
PATTERN   0.8662022E 00          13.54             13.50            220.0              0.563
```

```
EXPLORE  0.8662022E 00          13.54           13.50           220.0           0.563

EXPLORE  0.8662022E 00          13.54           13.50           220.0           0.563

EXPLORE  0.8652682E 00          13.54           13.50           215.4           0.568
PATTERN  0.8644297E 00          13.54           13.50           210.8           0.568
PATTERN  0.8644290E 00          13.54           13.50           210.8           0.568

EXPLORE  0.8644268E 00          13.54           13.50           210.8           0.570
PATTERN  0.8644268E 00          13.54           13.50           210.8           0.570

EXPLORE  0.8644268E 00          13.54           13.50           210.8           0.570

END OF 3 BLADE SEARCH...         72 FUNCTION EVALUATIONS PERFORMED

OPTIMUM PROPELLER :

NUMBER OF BLADES = 4
GEAR RATIO = 13.54
DIAMETER = 13.50 FT
ACTIVITY FACTOR = 220.0
INTEGRATED LIFT COEFFICIENT = 0.321
FUNCTION VALUE =  0.8630978E 00


TAKEOFF DISTANCE =  2209. FT
CP (STATIC) = 0.4719
CT/CP = 0.835
PROP EFFICIENCY (AT LIFTOFF) = .618
BLADE ANGLE (.75R) (AT V=0.) = 28.4 DEG

FOLLOWING VALUES ARE CALCULATED AT LIFTOFF :
 CP       J        M(EFF)      FT        FB        BHP      TJET

.472    0.8904    0.1678    1.0000    1.0043    4500.    483.


MAXIMUM SPEED = 397.6 KTS
BLADE ANGLE (.75R) = 53.8 DEG
PROP EFFICIENCY = .837

 CP       J        M(EFF)      FT        FB        BHP      TJET
```

```
CRUISE SPEED = 289.7 KTS
BLADE ANGLE (.75R) = 44.3 DEG
PROP EFFICIENCY = .849

 CP         J         M(EFF)       FT         FB          BHP        TJET

.324     2.1303     0.3980     1.0000     0.9611      1839.       146.


RANGE (AT CRUISE CONDITIONS AND CRUISE SPEED) =  1853. N MI  (45 MIN RESERVE)

ENDURANCE (AT CRUISE POWER AND CONDITIONS) =   6.2 HRS  (45 MIN RESERVE)


MAXIMUM RATE OF CLIMB = 3156. FT/MIN
CLIMB SPEED =  205. KTS
BLADE ANGLE (.75R) = 40.7 DEG
PROP EFFICIENCY = .755

 CP         J         M(EFF)       FT         FB          BHP        TJET

.472     1.5104     0.2951     0.9968     0.9522      4500.       379.


AT THIS POINT, IF YOU WISH TO TERMINATE PROGRAM OPERATION, TYPE "0."
IF YOU WISH TO RETURN TO AN INPUT STEP, TYPE NUMBER OF STEP
0.
IHC002I STOP       0
R; T=91.17/105.46 18.36.21
```

## LIST OF REFERENCES

1. Goldstein, S. "On the Vortex Theory of Screw Propellers", Proceedings of the Royal Society, A123 p. 440, 1929.

2. McCormick, B. W., Jr., Aerodynamics of V/STOL Flight, p. 73-99, Academic Press, 1967.

3. N.A.C.A. Technical Report 640, The Aerodynamic Characteristic of Full-Scale Propellers Having 2, 3, and 4 Blades of Clark Y and R.A.F. 6 Airfoil Sections, by Hartman and Biermann, 1938.

4. Hooke, R. and Jeeves, T. A., "Direct Search Solution of Numerical and Statistical Problems," Journal of the Association for Computing Machinery, v. 8, p. 212-229, April 1961.

5. Bell, M. and Pile, M. C., "Remark on Algorithm 178, Direct Search," Communications of the Association for Computing Machinery, v. 9, p. 684-685, September 1966.

6. Wilde, D. J., Optimum Seeking Methods, Prentice-Hall, 1964.

7. Generalized Method of Propeller Performance Estimation, Revision 'A' to PDB 6101, Hamilton Standard Division, United Aircraft Corporation, Windsor Locks, Connecticut, June 1963.

8. Dommasch, D. O., Sherby, S. S., and Connoly, T. F., Airplane Aerodynamics, 3d ed., p. 304-343, Pitman, 1961.

9. YP-3C/P-3C Orion Performance Characteristics with T56-A-14 Engines, LR22310, Lockheed California Company, Burbank, California

10. H. S. Specification No. HS 1845 Covering the 54H60-69 Propeller for the Lockheed P3V Airplane, HS 1845, Hamilton Standard Division, United Aircraft Corporation, Windsor Locks, Connecticut, August, 1959.

## INITIAL DISTRIBUTION LIST

| | | No. Copies |
|---|---|---|
| 1. | Defense Documentation Center<br>Cameron Station<br>Alexandria, Virginia 22314 | 2 |
| 2. | Library, Code 0212<br>Naval Postgraduate School<br>Monterey, California 93940 | 2 |
| 3. | Chairman, Department of Aeronautics<br>Naval Postgraduate School<br>Monterey, California 93940 | 1 |
| 4. | Professor D. M. Layton, Code 57Ln<br>Department of Aeronautics<br>Naval Postgraduate School<br>Monterey, California 93940 | 1 |
| 5. | LTJG Robert L. Shaw, USN<br>248 Greengate Lane<br>Spartanburg, South Carolina 29302 | 1 |
| 6. | Navy Plant Representative<br>Hamilton Standard Division<br>United Aircraft Corporation<br>Windsor Locks, Connecticut 06096 | 1 |
| 7. | Navy Plant Representative<br>Lockheed California Company<br>Burbank, California 91503 | 1 |

# INITIAL DISTRIBUTION LIST

| | | No. Copies |
|---|---|---|
| 1. | Defense Documentation Center<br>Cameron Station<br>Alexandria, Virginia 22314 | 2 |
| 2. | Library, Code 0212<br>Naval Postgraduate School<br>Monterey, California 93940 | 2 |
| 3. | Chairman, Department of Aeronautics<br>Naval Postgraduate School<br>Monterey, California 93940 | 1 |
| 4. | Professor D. M. Layton, Code 57Ln<br>Department of Aeronautics<br>Naval Postgraduate School<br>Monterey, California 93940 | 1 |
| 5. | LTJG Robert L. Shaw, USN<br>248 Greengate Lane<br>Spartanburg, South Carolina 29302 | 1 |
| 6. | Navy Plant Representative<br>Hamilton Standard Division<br>United Aircraft Corporation<br>Windsor Locks, Connecticut 06096 | 1 |
| 7. | Navy Plant Representative<br>Lockheed California Company<br>Burbank, California 91503 | 1 |

UNCLASSIFIED

Security Classification

# DOCUMENT CONTROL DATA - R & D

*(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)*

|  |  |  |
|---|---|---|
| ORIGINATING ACTIVITY *(Corporate author)*<br>Naval Postgraduate School<br>Monterey, California 93940 | 2a. REPORT SECURITY CLASSIFICATION<br>Unclassified<br>2b. GROUP | |

REPORT TITLE

Computerized Aerodynamic Optimization of Aircraft Propellers

DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Master's Thesis; June 1970

AUTHOR(S) *(First name, middle initial, last name)*

Robert Linford Shaw

|  |  |  |
|---|---|---|
| REPORT DATE<br>June 1970 | 7a. TOTAL NO. OF PAGES<br>101 | 7b. NO. OF REFS<br>10 |
| . CONTRACT OR GRANT NO.<br><br>. PROJECT NO.<br><br>:.<br><br>l. | 9a. ORIGINATOR'S REPORT NUMBER(S)<br><br>9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)* | |

. DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited.

|  |  |
|---|---|
| . SUPPLEMENTARY NOTES | 12. SPONSORING MILITARY ACTIVITY<br>Naval Postgraduate School<br>Monterey, California 93940 |

. ABSTRACT

The objective of this thesis was to develop a practical computer system for use of empirical data in the aerodynamic optimization of aircraft propellers. The system was designed for use with the IBM 2741 on-line computer terminal. This program provides instructions to the operator during execution, and allows interaction by the operator for input and alteration of data, and for program instructions.

The Lockheed P-3C aircraft was chosen as the subject for test and evaluation of the program. The currently operational propeller of this aircraft was tested to compare the program's prediction of aircraft performance against flight test information. An attempt was then made to select a propeller which would provide better performance under the same constraints as those imposed in design of the operational propeller.

D FORM 1 NOV 65 1473 (PAGE 1)

'N 0101-807-6811

UNCLASSIFIED

Security Classification

A-31408

| KEY WORDS | LINK A | | LINK B | | LINK C | |
|---|---|---|---|---|---|---|
| | ROLE | WT | ROLE | WT | ROLE | WT |
| Aircraft propeller | | | | | | |
| Aerodynamic propeller optimization | | | | | | |
| Computerized optimization | | | | | | |
| Aircraft propeller selection | | | | | | |

1 OCT 74 23107
26018

Thesis 118827
S43734 Shaw
c.1 Computerized aerodynamic optimization of aircraft propellers.

1 OCT 74 23107
26018

Thesis 118827
S43734 Shaw
c.1 Computerized aerodynamic optimization of aircraft propellers.

thesS43734
Computerized aerodynamic optimization of
3 2768 001 94399 6
DUDLEY KNOX LIBRARY